

Modal Considerations on Information Processing and Computation in the Nervous System*

Abel Wajnerman Paz

Universidad de Buenos Aires;

Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

abelwajnerman@gmail.com

We can characterize computationalism very generally as a complex thesis with two main parts: the thesis that the brain (or the nervous system) is a computational system and the thesis that neural computation explains cognition. As Piccinini and Bahar (2012) point out, over the last six decades, computationalism has been the mainstream theory of cognition. Nevertheless, there is still substantial debate about which type of computation explains cognition, and computationalism itself still remains controversial. My aim in this paper is to make two main contributions to the debate about the first sub-thesis of computationalism, i.e. that the brain is a computational system. First, I want to offer an accurate elucidation of the notion relevant for understanding computationalism (the notion of computation) and clarify the relation between computation and information as well as the relations between both computation and information processing and the nervous system. Second, I want to argue against a peculiar form of computationalism: the thesis that neural processes are *constitutively* computational in some sense; that neural processes cannot be realized by a system that is not in some sense computational. I will call this thesis “*modal computationalism*.” In particular, I want to argue that neural processing can be realized by a system that is not a *sui generis* computer (i. e.,

*I thank Gualterio Piccinini very much for his comments on many points of this paper and, mainly, for his inspiring work on computationalism. I am also grateful to the three referees from the Journal of Cognitive Science for their very useful comments. Finally, I want to thank Sergio Barbieris, Sabrina Haimovici, Mariela Destéfano, and especially Liza Skidelsky for discussing very exhaustively one of the latest versions of this paper and contributing to making the deep modifications necessary to arrive at this final version.

a computing system that is neither digital nor analog) and by a system that is not a generic computer (a computer in the most general sense: one that includes digital, analog, and any other kind of computation). *Actual* neural processing is presumed to be computational in these two senses (Piccinini and Bahar 2012). I will argue that, even if this is true, neural processing *can* be realized by a computing system that is not of the same kind as those that perform actual neural processing and even by a system that is not computational at all.

Key words: *semantic information, computationalism, feedback control, neural computation, generic computation*

1. Introduction

We can characterize computationalism very generally as a complex thesis with two main parts: the thesis that the brain (or the nervous system) is a computational system and the thesis that neural computation explains cognition. As Piccinini and Bahar (2012) point out, over the last six decades, computationalism has been the mainstream theory of cognition. Nevertheless, there is still substantial debate about which type of computation explains cognition, and computationalism itself still remains controversial. My aim in this paper is to make two main contributions to the debate about the first sub-thesis of computationalism, i.e. that the brain is a computational system. First, I want to offer an accurate elucidation of the notion relevant for understanding computationalism (the notion of computation) and clarify the relation between information and computation as well as the relation of computation and information processing to the nervous system. Second, I want to argue against a peculiar form of computationalism: the thesis that neural processing is *constitutively* computational in some sense; that neural processing cannot be realized by a system that is not in some sense computational. I will call this thesis *modal computationalism*. In particular, I want to argue that neural processing can be realized by a system that is not a *sui generis* computer (i.e. a computing system that is neither digital nor analog) and by a system that is not a generic computer (a computer in the most general sense, which includes analog, digital, and any other kind of computation). Actual neural

processing is presumed to be computational in these two senses (Piccinini and Bahar 2012). I will argue that neural processing can be realized by a computing system that is not of the same kind as the ones that perform actual neural processing and even by a system that is not computational at all. It is important to point out that, as we will see later, even if one is not interested in the forms of *modal computationalism* I will consider here, the elucidations of the relevant concepts that I will offer in the first part of the paper will be useful more generally to consider any form of computationalism.

Piccinini and Scarantino (2011) take on an important philosophical task in the context of this debate. The authors elucidate the concepts of computation and information, and they draw two main conclusions about how these concepts are related. First, they show that every notion of computation is different from every notion of information. Second, they then affirm that the three main notions of information (Shannon's non-semantic notion, natural semantic information, and non-natural semantic information) imply the generic notion of computation. This can be important in the debate, because if the neural processing is empirically established to be performed by an information processing system, it follows that neural processing is computational in the generic sense. This is the second argument for generic computationalism that Piccinini and Bahar (2012) offer. The first argument is based on the fact that the nervous system has the central feature of a generic computational system: medium independent vehicles. Piccinini and Bahar also argue against digital and analog computationalism. They posit that although the nervous system computes, the kind of computation it performs is neither digital nor analog, i.e., it is a *sui generis* kind of computation.

This paper has two main parts. In the first part (sections 2.1 and 2.2), I will consider the concepts of information processing and generic computation, the relation they have to each other, and the relation they have to the nervous system. In section 2.1, by comparing the notions of computation and information, I will show that computation is best characterized as the manipulation of vehicles that have what I call "restricted functional relevance." I will then show that information processing does not imply computation. In section 2.2, I will argue that information processing is

related to the nervous system in a stronger way than computation is. I believe that feedback control, the function peculiar to neural activity, implies information processing, but not computation. To argue for this thesis I will consider and criticize the elucidation of the notion of feedback control offered by Wimsatt (1971). I believe that even a very strong notion of feedback (such as Wimsatt's) does not imply computation and that even the much weaker notion of feedback I defend implies information processing.

With these conceptual tools, in the second part (sections 3.1 and 3.2), I argue against the types of computationalism that Piccinini and Bahar (2012) defend: generic and *sui generis*. In each of these cases, "computationalism" will be taken in a different sense than Piccinini and Bahar treat it. Therefore, my theses will be compatible with theirs. Piccinini and Bahar argue that *actual* neural processing is performed by *sui generis* and generic computers. I will argue that, even if we accept that position, neural processing *can be realized* in a system that is neither a *sui generis* nor a generic computer. In section 3.1, I show that feedback control is *constitutive* of neural processing and distinguish it, in this respect, from computation and information processing. This means that feedback control is the only one of these features (computation, information processing, and feedback control) that is needed for the realization of neural processing. The other features (computation and information processing) will be (*indirectly*) needed for the realization of neural processing only if they are necessary for the realization of a feedback control system. I will then argue that, since feedback control is compatible with digital and analog computation, *sui generis* computation is not constitutive of neural processing. Neural processing does not need *sui generis* computation to be realized. Thus, I maintain that a form of *sui generis* computationalism is false. In section 3.2, I will argue that since feedback control can be realized even by a system that is not computational at all, as feedback control does not need generic computation to be realized, computation in the generic sense is not constitutive of neural processing. A form of generic computationalism is also false. The peculiarity of these forms of computationalism is that they are answers to *modal* questions about the relation between computation and the neural processing, questions that lead us to consider the range of

multiple realizability of these kind of processes.

2. Computation, Information Processing, and the Nervous System

2.1. Computation and Information Processing

Piccinini and Scarantino (2011) distinguish three main notions of information: a non-semantic, a natural semantic, and a non-natural semantic notion. The notion formalized by Shannon (1948) is non-semantic in that a piece of information in his sense is fully characterized by two features: it is a physical structure distinguishable from a set of alternative physical structures, and it belongs to an exhaustive set of mutually exclusive selectable physical structures with well-defined probabilities. No semantic property is involved in its individuation. However, an information theorist could be also interested in the specific content of a given piece of information. Piccinini and Scarantino (2011) borrow Grice's (1957) distinction between natural and non-natural meaning and apply it to semantic information. A case of natural meaning could be exemplified by the sentence, "those spots mean measles," which is true only in the case that the patient has measles. Non-natural meaning is exemplified by a sentence like "those three rings on the bus bell mean that the bus is full," which is true even if the bus is not full. Similarly, we classify information as natural when there is a physical, reliable correlation between two states. Spots carry information about measles only when the spots are reliably correlated with measles. On the contrary, the ringing of the bus bell carries information about the bus being full by virtue of a convention.

So, given this characterization of semantic information, what determines whether an object o or event e is a token of a certain informational vehicle type I ? The obvious answer is that whether an event e or object o belongs to an informational type I depends on the kind of information it carries. But the information that an object carries, as we have seen, is fully determined by the reliable correlation that the object or event has with tokens of an object or event type T or by the convention that links the object and the tokens of T . So, we can say that an object o or event e belongs to an informational type I if and only if it has a reliable correlation with tokens of an event type T or there exists a convention that links o

or e with tokens of T . But is there something else that is necessary for an object to belong to an informational type I ? Particularly, do the structural or intrinsic properties of o or e play some role in the determination of their informational type? I believe that, since the correlation of an object with T s or the convention that links that object with T s is a sufficient condition to belong to I , structural properties of o or e will be necessary for belonging to I only if those structural properties of o or e are necessary for having a correlation with tokens of T or forming a convention that links o or e with T s. However, I think that this is not the case. Two events that share no structural properties can be correlated with the same event type and thereby carry the same information. For example, for a predator, the specific scent s of prey p that reaches its olfactory system and some pattern of light l that reaches its visual organs from p share no structural properties, but both can be reliably correlated with (and thereby carry information about) the presence of p . Both events can belong to the same informational type I (the type of informational vehicles that carry information about the presence of p) although they share no structural property. Another ordinary example of this feature of natural semantic information could be this: smoke and my concept FIRE do not share any structural property functionally relevant for the reliable correlation they both can have with fire. The fact that semantic non-natural information has the same feature is more obvious, given that conventions can be more or less arbitrary. What is important about this phenomenon (i.e., that vehicles that carry the same information and belong to the same *informational* type can share no structural properties) is that it shows the multiple realizability of informational vehicles. Vehicles of information processing can be realized in *completely* different physical bases, i.e., physical bases that have no common intrinsic or structural property. I think this feature is important in comparing vehicles of information processing with vehicles of computation.

The more general notion of computation, the notion of generic computation, is introduced by Piccinini and Scarantino as a way to capture “all the relevant uses of ‘computation’ in cognitive science” (2011, p. 10). Piccinini and Scarantino affirm that “generic computation is the processing of vehicles according to rules that are sensitive to certain vehicle properties and, specifically, to differences between different portions of the vehicles”

(2011, p. 10). Piccinini and Bahar (2012) characterize the rules that define generic computation more specifically:

“the rules (i.e., the input-output maps) that define a computation are sensitive only to differences between portions of the vehicles along specific dimensions of variation—they are insensitive to any *more concrete physical properties* of the vehicles” (pp. 6-7)

Piccinini and Scarantino call this kind of vehicles “medium independent” because this characteristic property makes possible the multiple realizability of computational vehicles, i.e., that a given computation be implemented in different physical media (mechanical, electromechanical, electronic, magnetic, etc.). A computational vehicle type can be realized in a variety of media because very specific or concrete physical properties are not necessary for its realization.

I believe that there is a property that informational and computational vehicles share, and I will follow Piccinini and Bahar in calling it “medium independence.” But I also believe that this property is not identical to the defining property of computational vehicles. So, I will use the term “restricted functional relevance” to refer to the defining property of computational vehicles and keep the term “medium independence” to refer to the property that I believe both kinds of vehicle share. I believe that “restricted functional relevance” better captures the more specific property of computational vehicles and that “medium independence” better captures the more general one (the one I think it is shared by computational and informational vehicles). I will maintain that computational and informational vehicles are similar only in that they do not depend on a particular physical medium for their realization, i.e., in that they both have multiple realizability. So, I will use “medium independence” to refer only to multiple realizability (knowing that this a wider sense than the one given by Piccinini and Bahar to the same expression). But, I will affirm that while computational vehicles exhibit this multiple realizability because their intrinsic functionally relevant properties are abstract, informational vehicles have multiple realizability because their defining property is relational, not intrinsic. So, by “restricted functional relevance” I mean only that the

functionally relevant properties of computational vehicles, according to Piccinini and Bahar's definition, are not very specific or concrete. I believe that Piccinini and Bahar are correct in thinking that both being a vehicle of information processing and having what I refer to as "restricted functional relevance" conceptually imply having multiple realizability. Nevertheless, I will argue that being informational and having restricted functional relevance are not extensionally equivalent concepts. So, information processing does not imply computation.

We have seen that, since a vehicle of informational type I is fully characterized by some relational property its tokens have with tokens of states or events of type E (be it a causal correlation or a convention), *no* particular structural property is required for the realization of the tokens of I . Two states that share no structural properties can both be tokens of I if they correlate with tokens of type E events (or if they are linked by a convention with type E events). We saw this with the visual/olfactory prey detection and smoke/FIRE examples above. I also said that it is this feature of informational vehicles that makes them multiple realizable or medium independent. On the other side, the feature of computational vehicles that make them medium independent is not their relational character, but rather the fact that the features that define them are not very specific or concrete. Computational vehicles can be realized in different physical bases because very abstract properties of them are functionally relevant. For example, for a component of a system to be a token of an analog computation vehicle type, the functionally relevant states of that component only need to be continuous. To be a token of a digital computation vehicle type, the functionally relevant aspects need to be discrete. For the component to be a token of a specific digital or analog vehicle type, even more specific structural properties will be required. However, as there will be many other aspects of these tokens that will not be functionally relevant, they will be medium independent. On the contrary, we have seen that the realization of informational vehicles does not impose any restriction on their specific structural properties. In particular, it is not required that the functionally relevant aspects of an informational vehicle token v be restricted to more abstract aspects of v . Any informational vehicle type can be realized in a token that has no restricted functional relevance, whose

functionally relevant properties are very specific or concrete. Therefore, being informational and having restricted functional relevance are not extensionally equivalent concepts.

I said above that restricted functional relevance was the common feature of computational vehicles. The medium independence of vehicles of both digital and analog computation follows from the fact that not very specific properties of them are functionally relevant. If being informational and having restricted functional relevance are concepts that are not extensionally equivalent but that both imply medium independence, then the concepts of medium independence and restricted functional relevance are not extensionally equivalent either (the notion of medium independence will apply to token vehicles that have no restricted functional relevance; namely, some informational vehicle tokens). We therefore cannot define computational vehicles as medium independent, but rather only as vehicles of restricted functional relevance. I think that Piccinini and Scarantino, in defining generic computation, identified the correct property, yet they failed to characterize it appropriately. They realized that what I call restricted functional relevance was the defining property of generic computation, but they characterized it as medium independence, *as the property shared by computational and informational vehicles*. With this in mind, we can further conclude, more importantly, that because information processing does not imply restricted functional relevance, and since computation must be defined as the manipulation of vehicles of restricted functional relevance, then information processing does not imply computation.

To summarize, I have argued that medium independence and restricted functional relevance are different concepts. The notions of restricted functional relevance and being informational are not extensionally equivalent, but they both imply medium independence. This means that medium independence (understood as multiple realizability) does not imply restricted functional relevance, and therefore we cannot define computational vehicles as medium independent, but only as vehicles of restricted functional relevance. From this it follows, contrary to the affirmations of Piccinini and Scarantino, that *medium independence of informational vehicles does not imply their computational character*. In the next section, I will examine the distinction between information

and computation considering not those concepts themselves, but rather their relation to the concept of feedback control, the main function of the nervous system. As I stated above, this will be important later (in sections 3.1 and 3.2) to argue against a form of generic and *sui generis* computationalism. As I will argue later that feedback control is constitutive of neural processing (that it is the most adequate criterion for assessing the range of its multiple realizability), the conceptual relations that feedback control has with computation and information processing (as the conceptual relation established in this section between computation and information processing) will be relevant to determine whether computation itself is constitutive of neural processing.

2.2. Computational and Informational Feedback Control

In the previous section, I offered an elucidation of computation and argued against the implication from information processing to computation. This makes the line between computation and information processing sharper than it seemed in Piccinini and Scarantino's explanation. In this section, I will further distinguish computation and information processing in terms of their relation to the nervous system. I will argue that information processing, but not computation, is constitutive of feedback control, the function peculiar to neural activity. In particular, I think that even a very strong notion of neural feedback does not imply computation and that even a very weak notion of neural feedback implies information processing.

Feedback control is the function that is specific to neural activity. The specific function of neural activity differentiates the nervous system from other functionally organized systems. The nervous system belongs in the class of feedback control systems together with autopilot systems, for instance. Piccinini and Bahar (2012) note that the thesis of the nervous system as a feedback control system is not trivial and took serious scientific work to establish rigorously (Cannon 1932). Nevertheless, by now it is uncontroversial, though there is still substantial debate on the nature of feedback control itself. Piccinini and Bahar characterize nervous feedback saying that the nervous system controls the behavior of an organism in response to stimuli coming from both the organism's body and the environment, *including the effects of the nervous system itself*. So,

according to this characterization, to execute feedback control over a larger system S of which it is a part, a system S_I only needs to cause states in S (control the behavior of S) in a way that depends on previous states of S itself or its environment (in response to stimuli coming from them) that are previous effects of S_I itself. Thus, if we have feedback control of a system S_I over a system S , there will be a reliable causal correlation between previous states of S or its environment caused by S_I , internal states of S_I , and new states caused in S by S_I .

I believe that this characterization of feedback control is basically correct. What is constitutive of feedback is the causal loop I have just described between previous effects of the feedback systems and new effects. I think that this is all we need for a system to execute feedback control. Nevertheless, in light of recent discussion on the notion of feedback, I feel I need to offer a defense of this characterization. The notion presented by Wimsatt (1971)¹ is much more demanding than Piccinini and Bahar's. First, I will argue that even this stronger notion does not imply computation (neither *sui generis* nor generic). Second, I will defend the weaker notion presented above. Finally, I will argue that even this weaker notion implies information processing.

Before entering into the debate about feedback control, there is a further constraint for *neural* feedback control that is absent in Piccinini and Bahar's characterization and that we should emphasize. Neural feedback control is negative. Very roughly, feedback control is negative when the value of the output signal is diminished by the system when the value of the same variable goes beyond a certain threshold. In this section, I will be concerned exclusively with characterizations of this notion of negative feedback.

There has been substantial debate about the nature of feedback control. I believe that Wimsatt (1971) has contributed two ideas that are very relevant for thinking about the relation between feedback, computation, and information. Wimsatt (1971) argues that feedback control cannot be characterized solely in behavioral terms, that is, in terms of a certain correlation between specific inputs and outputs of a system. This means that

¹ I am in debt to a referee of the Journal of Cognitive Science for encouraging me to examine my theses on the light of Wimsatt's stronger notion of feedback control.

to determine if a system executes feedback control, one has to determine, to some extent, the nature of its internal structure. This is important for our purposes, because whether feedback control implies computation depends exactly on how strong those constraints on the internal structure of feedback control systems are. Wimsatt (1971) also argues that two very widely accepted conditions to characterize the internal structure of a feedback control system are not sufficient. Nevertheless, he accepts them as necessary. In what follows, I will advance three theses. First, I will argue that even the notion of negative feedback that includes the constraints on the internal structure of a negative feedback system that Wimsatt mentions as necessary conditions does not imply computation (neither *sui generis* nor generic). Secondly, I will argue that those constraints are not even necessary for negative feedback. I will defend a weaker notion of negative feedback. Lastly, I will argue that even this weaker notion implies information processing.

Wimsatt considers three constraints that have been widely accepted for feedback control systems. The first two appear in the elucidation of feedback control given by authors such as Beckner (1959), Ashby (1960), Sommerhof (1950), and Nagel (1961). They claim that a feedback control system requires: (1) power to compensate for environmental changes that might impede the system's progress towards the goal; and (2) independent variables that define the system and its environment. It seems clear that the second condition does not imply that feedback control vehicles must be computational. It simply requires that there be two logically independent, but causally related, variables in the system in addition to environmental variables. This second condition puts no restriction on the specific nature of those variables (they could be discrete or continuous, medium-dependent or medium-independent, etc.). Thus, this condition alone makes it possible for vehicles of feedback control to be instantiated in a non *sui generis* computational system or even in a system that is not computational at all. The first condition, the one that specifies the compensation of negative feedback, determines a more specific task for those variables. We will come back to this in a moment.

The third condition is that a system be capable of reaching the same set of equilibrium values for its state variables from an arbitrarily large range

of different initial values of the state variables (this is the primary feature of Braithwaite's (1953) analysis). Von Bertalanffy called this phenomenon 'equifinality,' and both he and Kacser (1957) have given formal analyses of this phenomenon for open systems of the general type which Wimsatt claims are not feedback systems. So, although this feature could be necessary for feedback control, it is not sufficient.

The first condition, the 'compensatory' phenomenon frequently invoked, states that changes in one (or more) state variable(s) will be partially or wholly compensated for by changes in one or more other state variables in such a way that the original perturbed variable(s) will be wholly or partially restored to their 'normal' equilibrium values. This phenomenon has been described by Kacser (1957) as 'buffering,' a word which is a better clue to its real significance than those employed in the analyses mentioned above. The first and third features are clearly related. Equifinality describes the effect of compensation. The compensation condition gives the mechanism by which this 'equifinality' of non-equilibrium states is brought about: some variables 'take the load' from the 'stressed' variables and allow them to partially or completely approach or regain their 'normal' values.

Wimsatt shows that the condition of compensation, like the condition of equifinality, could be necessary for feedback control, but is not sufficient. He describes two mechanisms that satisfy the compensation condition, but the first of them is not a feedback control system. The feature that the second has, but not the first does not, is a further necessary condition for feedback control that Wimsatt introduces. I will claim that neither compensation nor equifinality are necessary for feedback control, and that the additional condition that one should extract from Wimsatt's model is just the causal loop that we described above (the causal loop that characterizes a *negative* feedback system).

Before arguing for the weaker notion of negative feedback control, I want to argue that even these stronger requirements do not imply computation. The condition of equifinality requires that the internal variables of the system be able to reach an equilibrium value from an arbitrarily large set of values. The condition of compensation requires that this must be done by making other variables take the load of the stressed variables. The variables that are above a certain threshold must go back to a value below that

threshold (their equilibrium value) by raising the values of other variables. It can easily be seen that these restrictions do not imply that negative feedback vehicles should be non-digital or non-analog like actual neural vehicles. The capacity of a variable to go back to an equilibrium value and the capacity of a variable to take the load of another variable (increase its value if the value of the other decreases) imposes no restriction to the nature of those values. *It only requires that certain changes of value of the variables be possible and that certain correlations between the values of the variables be given.* However, this leaves open the possibility that the values themselves may be continuous or discrete. Neural feedback control therefore does not imply *sui generis* computation. Furthermore, these restrictions also leave open the possibility that the functionally relevant aspects of the variables may be very specific or concrete. Therefore, neural feedback control does not imply restricted functional relevance. Neural feedback does not imply computation. Nevertheless, I think that the notion of feedback does not even require these two structural requisites of equifinality and compensation. This will make the two points made about feedback and computation even more obvious.

Wimsatt presents two hydrodynamic models involving the flow of water into and out of a pair of interconnected tanks. The non-feedback system consists of two geometrically congruent tanks of constant cross-section interconnected by an orifice with an input of water at the top of one of the tanks at a specific rate r_1 and with an output magnitude r_0 in an orifice at the bottom of the other tank. Water passes through the connecting orifice at another specific rate r_s (that is positive if flow moves from the first tank to the second). r_s is a function of the water level of the two tanks (l_1 and l_2), the gravitational force g , and the area of the orifice that connects them. The condition of equilibrium is given in the situation in which the values of l_1 and l_2 are proportional to each other, they are proportional to r_0 , to r_1 , and to r_s , and these three values are equal. In the situation in which the value of l_2 is superior to the value of l_1 , r_s will be negative, although r_0 and r_1 will remain positive. The flow of water from one tank to the other will reverse until l_1 and l_2 reach their equilibrium values. When they do, r_s will go back to its equilibrium value too. This simple open system thus exhibits both of the features supposedly sufficient for a feedback system: (1) the system

tends to approach equilibrium values from arbitrary non-equilibrium states, and (2) the system 'buffers,' as an unusually high value of the variable of second tank has been partially compensated for by an increase in the variable in the first tank.

Nevertheless, the second model will show why this first is not a feedback control system. In the second model, we again have two interconnected tanks, but there is also a pressure sensor at the output r_0 of the second tank that detects changes in r_0 (and, as the two variables are connected, it also detects changes in l_2) and is connected to a variable input valve that controls r_1 in such a way that deviations from the equilibrium values of r_0 and l_1 are counteracted by changes in r_1 .

Wimsatt holds that the main difference between these two systems is that in the first, the compensation between the two internal variables depends on the possibility that the flow from the first variable to the second being reversible. In the second model, compensation can take place without this reversible character. I believe that it is another difference which determines that only the second is a feedback control system. Recall that a very intuitive required feature of feedback control system (the one that made us, through an argument that Wimsatt (1971, pp. 244-246) offers, reject the behavioristic account of feedback) is that there must be some kind of 'causal loop.' There must be a causal relation between the actual state of the output variable of the system and previous states of that same variable. That is why Wimsatt showed that we cannot give a behavioristic criteria for feedback control. If any input-output matrix can be reproduced by a net that has no loops, then, although we may find a system in which there is a *correlation* between values in the output variable and previous values of the same variable, this could be done without there being any causal relation between those values. In the case of the second pair of tanks, we find just this kind of loop required for feedback control. When a certain value at the output variable r_0 is given, this causes a change in the input variable r_1 which, in turn, changes the value of l_1 . The decrease of the input flow of water r_1 , given that the flow rate r_s from the first tank to the second remains constant, causes a decrease of the level of water in the first tank, l_1 . This, in turn, causes a decrease in the rate r_s of the flow from the first to the second tank (given that this rate is, in part, a function of the level of the first tank,

l_1). The decrease in r_s (given that the output r_0 still has the same value) causes a decrease in the level of the second tank, l_2 , and this finally causes a decrease in the value of r_0 , whose original value initiated this result.

In the first pair of tanks we simply do not have a causal loop like this. It is not the value at r_0 in a past moment that causes the decrease of the value of r_0 at a present moment. When r_0 has certain value, this depends on the level of water of the second tank, l_2 , and the value of *this* variable causes the decrease in the value of r_s which, in turn, causes a decrease of the value of l_2 . This decrease in l_2 causes a decrease in r_0 . So, a value of l_2 at a moment t_0 has two different causal paths: one of them leads directly to a certain value of r_0 at a time t_1 . The other leads to a modification of l_2 itself at t_2 (through the modification of r_s at t_1). This new value of l_2 at t_2 leads, in turn, to a new value for r_0 at t_3 . But the first value of the output variable r_0 at t_1 is not involved in the production of its new value at t_3 . So, here we have a case where there is only a correlation, but not a causal relation between actual states of the output variable of the system and its previous states. I believe that this feature, which the first system lacks, is what is necessary for feedback control.

Furthermore, I believe that this model shows that one of the other conditions supposedly required for feedback control, compensation, is not necessary. It must be noted that the second system is a feedback control system according to Wimsatt, but no compensation occurs between its internal variables. The decrease of the value of r_0 is caused by the *decrease* of the values of *all* the other variables (first r_1 , then l_1 , r_s , and finally l_2). There is no variable 'taking the load' from the high value of r_0 . Therefore, compensation is not necessary for feedback control.

Additionally, I believe that the form of equifinality required is much weaker than the one described. In particular, the equifinality I believe we should demand does not imply anything about the internal structure of the system. Recall that the equifinality of a system, as described, is that it is capable of reaching the same set of equilibrium values for its state variables from an arbitrarily large range of different initial values of the state variables. Restricted just to an output variable, such as Wimsatt's r_0 , this is simply part of the description of the causal loop required for negative feedback. This system is a negative feedback system because r_0 can go

from an arbitrarily large range of states back to its equilibrium value. However, it seems that the condition of equifinality requires that *all* the internal variables of the system have this property. I think that this is false. Why should we require this for negative feedback? I believe that in many systems (such as the two models presented by Wimsatt), this is the only way that we can have the kind of causal loop required for the output variable. In both systems, when the output variable r_0 goes beyond a threshold T , the other variables must change their values (some must increase their values, other reduce it, depending on the system) until the output variable has reached its equilibrium value E . But when r_0 has reached E , all the other variables in Wimsatt's systems must go back to their previous value *so that r_0 stops decreasing*. Nevertheless, this is not necessarily so. A system can be a feedback system, having an output variable with the relevant kind of causal loop, without its internal variables having the capacity to go back to a certain value from an arbitrarily large set of values.

Suppose there is a system S that consists of two tapes, A and B , a device $D1$ that writes or erases symbols on A , a device $D2$ that writes symbols on B , and a sensor S that detects the number of letters on A (SA) and the number of letters on B (SB) and that is connected to $D1$ and $D2$. $D1$ keeps writing symbols on A unless $D2$ writes symbols on B . Furthermore, each time $D2$ writes a symbol on B , this causes $D1$ to erase a symbol on A . When S senses that SA has passed a certain threshold T , it makes $D2$ start writing symbols on B , and thereby makes $D1$ erase symbols from A until SA reaches certain value V below T . V is the equilibrium value for the variable SA of the system. SA reaches V through the modification (specifically, the increase) of the value of the other relevant variable (SB). But SB has no equilibrium value. The value of SB will keep rising each time SA passes T . Nevertheless, this system is clearly a negative feedback control system. There is an output variable SA of S that can go beyond a value T , and when this happens, it causes an internal process that culminates with the decrease of the value of SA to V . This is the kind of causal loop required for negative feedback, and it does not require equifinality.

Therefore, the only requisite necessary for feedback control that remains is that which appeared in Piccinini and Bahar's characterization: the outputs of the system must depend causally on previous effects of the system itself.

And since neural feedback is negative, we can add the restriction that the output must be an equilibrium value of a variable caused by previous non-equilibrium values of the same variable. As this characterization puts even less restrictions on the internal states of the system, it seems clear that it does not imply *sui generis* or generic computation. It only requires that those internal states correlate the actual equilibrium value of the output with the previous non-equilibrium ones.

We can now consider the relation between feedback control and information processing. I will argue that even the weak notion of feedback that I defend implies information processing. This relation between feedback control and information processing is conspicuously absent in Piccinini and Bahar's work. They only claim that the complexity of the feedback control performed by the nervous system makes it necessary for the nervous system to process internal variables that carry information about external variables:

[I]n order to exert sufficiently complex control functions, the only efficient way to perform feedback control is to possess and process internal variables that correlate reliably with relevant external variables. These internal variables must be processed based on the natural information they carry. (2012, p. 4)

They do not elaborate, however, on why the complexity of feedback control makes information processing necessary, nor do they comment on whether the necessity is conceptual, empirical, or of some other nature. In what follows, I will claim that the weaker notion of negative feedback control *conceptually* entails information processing.

According to the weaker notion defended, there must be *at least* two internal states of a negative feedback control system. The negative feedback loop requires one state S1 that detects a deviation of the output variable O from its equilibrium value E and causes O to decrease or increase, and another state S2 that responds to O being at E and causes the decrease or increase of the value of O to stop. But this is all that is required to be an information processing system: that internal states carry information (S1 about not-E and S2 about E) and that states that carry different information

have different causal roles so that we can talk about the system responding to the information of its vehicles (S1 causes the variation of O's value and S2 causes the variation to stop). So, if a system is a feedback control system, then it is an information processing system.

3. Modal Computationalisms

Piccinini and Bahar (2012) defend two forms of computationalism: generic and *sui generis* (neither digital nor analog). They defend computationalism as the thesis that the nervous system computes in the generic and in a *sui generis* sense. But there is another question we can ask regarding the relation between computation and the nervous system. We know that actual neural computation, spike train processing, can be realized in very different physical bases (neural tissue or silicon based circuits, for example) (Craver 2010). We therefore know that the specific physical basis in which actual neural computers are realized is not constitutive of them. The fact that the neural processes have certain functionally relevant feature *F*, among others, does not imply that it cannot be realized in a system that has the other features, but not *F*. If the nervous system can be realized that way, then *F* is a contingent feature of it. If not, it is a constitutive one. We can ask the same question of realizability not regarding the relation between neural processing and its specific physical basis, but regarding the relation between neural processing and neural computation itself. Can neural processing be realized in a basis that is *not* a *sui generis* (neither analog nor digital) computer, the kind of computing system that (according to Piccinini and Bahar) performs actual neural processing? A negative answer to this question can constitute a form of *sui generis* computationalism: the thesis that *sui generis* computation is constitutive of neural processing. In section 3.1, I argue that this thesis is false. There is also a second question we can ask regarding neural computation. Even if it turns out, as I believe it will, that neural processing can be realized in a system that is not a *sui generis* computer, we can still ask whether it can be realized in a system that is not computational at all. A negative answer to this question can constitute a form of generic computationalism, the thesis that computation is constitutive of neural processing. In section 3.2, I argue that this thesis is

also false.

3.1. Constitutive *sui generis* Computation

Piccinini and Bahar (2012) argue for *sui generis* computationalism, the thesis that neural computation is neither digital nor analog, but *sui generis*. Let us first see how we can define analog and digital computation. Roughly speaking, abstract digital computation is the manipulation of strings of discrete elements of finitely many types. Here we want to determine what kind of computation, if any, *can* be realized in the nervous system. Thus, we are interested primarily in concrete digital computation, or physical digital computation. These discrete elements may be physically implemented by what we call 'digits.' In a first approximation, concrete digital computation is the processing of sequences of digits according to general rules defined over the digits (Piccinini 2007). A digit, the atomic vehicle of concrete digital computation, is a macroscopic state (of a component of the system) whose type can be reliably and unambiguously distinguished by the system from other macroscopic types. A large number of possible microscopic states corresponds to each (macroscopic) digit type. For instance, a huge number of distinct arrangements of electrons (microscopic states) correspond to the same charge stored in a capacitor (macroscopic state). Artificial digital systems are engineered so as to treat all those microscopic states in one way: that which corresponds to their (macroscopic) digit type. No further physical properties of a physical medium are relevant to whether they implement digits. Thus, digits can be implemented by any physical medium with the right degrees of freedom. This is why Piccinini and Bahar (2012) maintain that vehicles of digital computation have what they take to be the common feature of computational vehicles: medium independence.

Digital computation is often contrasted with analog computation. Piccinini and Scarantino (2011) observe that the clearest notion of analog computation is that of Pour-El (Pour-El 1974, Rubel 1993, Mills 2008). According to this notion, abstract analog computers are systems that manipulate continuous variables to solve certain systems of differential equations. Continuous variables are variables that can vary continuously over time and take any real values within certain intervals. As digital

abstract computers, analog computers can also be physically implemented, and physically implemented continuous variables are different kinds of vehicles than strings of digits. While a digital computing system can always unambiguously distinguish digits and their types from one another, a concrete analog computing system cannot do the same with the exact values of (physically implemented) continuous variables. This is because the values of continuous variables can only be measured within a margin of error. A major consequence is that analog computations (in the present, strict sense) are a different kind of process than digital computations. While digits are unambiguously distinguishable vehicles, concrete analog computers cannot unambiguously distinguish between any two portions of the continuous variables they manipulate. Since the variables can take any real values, but there is a lower bound to the sensitivity of any system, it is always possible that the difference between two portions of a continuous variable is small enough to go undetected by the system. From this, it follows that the vehicles of analog computation are not strings of digits. Nevertheless, analog computations are only sensitive to the differences between portions of the variables being manipulated to the degree that they can be distinguished by the system. Any further physical properties of the media implementing the variables are irrelevant to the computation. Like digital computers, therefore, analog computers operate on medium-independent vehicles.

Piccinini and Scarantino (2011) point out that, in recent decades, the analogy between brains and computers has taken hold in neuroscience. Many neuroscientists have started using the term ‘computation’ for the processing of neuronal spike trains, that is, sequences of spikes produced by neurons in real time. The processing of neuronal spike trains by neural systems is often called ‘neural computation.’ Taking the types of computation that Piccinini and Scarantino have distinguished, there are two main questions that we can ask regarding neural computation:

- (1) Is the brain a computer in the generic sense?
- (2) If the brain is a computer in the generic sense, is neural computation best regarded as a form of digital computation, analog computation, or something else?

Piccinini and Bahar (2012) argue that the nervous system is neither an analog nor a digital computer. Since they also argue that the nervous system is computational in the generic sense, it follows that it is a *sui generis* computer. Let's briefly consider the main theses of analog and digital computationalism. Analog computationalism is committed to the view that the functionally significant signals manipulated by the nervous system are continuous variables and that the main tool for processing neural signals are integrators. Piccinini and Bahar (2012) point to several aspects of neural processing that suggest that it could be some kind of analog computation. Brains and analog computers both appear to have an essentially continuous dynamic, that is, their vehicles must be assumed to vary over real time in order to construct successful scientific theories of them (for the case of brains, see Dayan and Abbott 2001, Ermentrout and Terman 2010). Neuronal inputs—i.e., neurotransmitters and hormones—are most usefully modeled as continuous variables. In addition, their release and uptake is modulated by chemical receptors that operate continuously in real time. Also, neuronal dendrites and at least some neuronal axons transmit graded potentials—that is, more or less continuously varying voltage changes. In these respects, Piccinini and Bahar agree that brains appear to be more similar to analog computers than to digital ones.

However, they note that there is at least one crucial disanalogy between brains and analog computers:

Typically, if the total graded input signal received by a neuron is above a certain threshold, the neuron fires a spike. If the total signal received by a neuron is below the threshold, no spike is fired. As a consequence of the all-or-none character of the spike, computational neuroscience focuses on firing rates and spike timing as the most significant functional variables at the level of circuit and network. Thus, while firing threshold is subject to modulation by continuous variables (e.g., ion concentrations), firing rates vary in a graded way, and spike timing (in real time) may be functionally significant, none of these aspects of neural signals amounts to the use of continuous variables as vehicles within analog computers. (2011, p. 15)

On the other hand, digital computationalism can be traced back to the work of McCulloch and Pitts. They attempted to explain cognitive phenomena in terms of putative neural processes. McCulloch and Pitts (1943) introduced concepts and techniques from logic and computability theory to model what is functionally relevant to the activity of neurons and neural nets in such a way that (in our terminology) a neural spike can be treated mathematically as a digit, and a set of spikes can be treated as a string of digits. McCulloch and Pitts's empirical justification for their theory was the similarity between digits and spikes. Spikes *appear* to be discrete or digital, i.e., to a first approximation, their occurrence is an unambiguous event relative to the functioning of the system. This was the main motivation behind McCulloch's identification between spikes and atomic mental events, leading to his formulation of digital computationalism (Piccinini 2004).

Piccinini and Bahar (2012) offer exhaustive considerations against the possibility of typing spikes (or even spike rates) as digits and describing spike trains as strings of digits. Their considerations go so far as to consider a breadth of relevant theoretical possibilities and are therefore too extensive to be presented here. Since the thesis I will defend is independent of the one that these considerations support, it is not necessary to assess them. I said earlier in this section that there are two questions regarding neural computation given Piccinini and Scarantino's elucidation. The first question is about neural processing as generic computation and, as we will see in the next section, Piccinini and Bahar answer positively. Thereby, the second question is, given a positive answer to the first question, about the specific kind of computation performed by the nervous system. Piccinini and Bahar argue against the digital or analog character of neural computation. They affirm that neural computation is *sui generis*. I will argue that although neural processing is performed actually by a *sui generis* computer, this is not constitutively so. In other words, neural processing can be realized by either a digital or an analog computer.

In order to argue for or against a thesis about the constitutive character of a feature of neural processing, one has to find a way to distinguish between constitutive and contingent features. Regarding different properties of action potentials, we have the criterion of functional relevance. Spike rates and timing are the variables correlated with different causes and effects

inside and outside the nervous system, and they do not need to be realized in neural tissue. I believe that there is also such a criterion regarding the processes performed by different systems of a biological organism and, in particular, regarding different features related to the processes performed by the nervous system and discussed by Piccinini and Bahar (2012). I will claim that feedback control is constitutive of neural processing in a way that the different kinds of computation and information processing are not. I showed, in section 2.2, that feedback control entails information processing. So, information processing will be also constitutive of neural processing (although indirectly). Nevertheless, I will show that feedback control does not entail *sui generis* computation, and so *sui generis* computation is not constitutive of neural processing.

Piccinini and Bahar (2012) do not seem interested in computationalism as a thesis about the constitutive character of a feature of the neural processing. They argue for the thesis that the nervous system computes in some sense without considering the possibility that neural processing be realized in a system that is not computational in that same sense. I will return to this aspect of Piccinini's position later. What is important to acknowledge here is that the two forms of *sui generis* computationalism have some independence. If it turns out, as I think it will, that *sui generis* computation is not constitutive of neural processing, there still remains the fact established by Piccinini and Bahar that the nervous system is *in fact* a computer of this kind.

The peculiar function discovered in the nervous system, negative feedback control, has something in common with the main functions of the other systems of human organisms. This similarity, however, is not shared by other features ascribed to the nervous system, such as being computational or being an information processing system. To execute negative feedback control over larger system S of which it is a part, a system S_I (1) must cause states in S (control the behavior of S) in a way that depends on previous states of S itself or its environment caused by S_I , and (2) the output of S_I (the behavior of S) must reach an equilibrium value from a process that causally depends on previous non-equilibrium values of S_I 's output itself. So, the feedback control function, as the main function of other systems in human organisms, determines a way in which a system relates to the organism of

which is a part and to whose behavior it contributes. Thus, feedback control is constitutive of the nervous system (and of neural processing) in a way that information processing and computation are not.

The fact that the nervous system is a computational or an information processing system says nothing about the way that system will interact with the rest of the organism or contribute to its behavior. Regarding information, as we have seen, we only need their states to carry information and the states that carry different information to have different causal roles. But those roles do not determine some specific output; there is no specific way an informational system must contribute to the behavior of a larger system it forms a part of. Regarding computation, the same point applies: as long as a computational system manipulates vehicles of restricted functional relevance, there is no specific output that every computational system that is part of a larger system should have. On the contrary, to characterize a system as a feedback control system, one needs to characterize the causal correlation between values of its output, i.e., values of the variable that is connected with what is beyond the system. To characterize something as a feedback system, we have to specify the way it interacts with what is beyond it. It is because of this that feedback control is the only of these three features (computation, information processing, and feedback control) that we can demand for the realization of a nervous system or a neural process. Feedback control is the only one of these three features that characterizes the nervous system as a *system of a given organism* (and neural processes as processes of a system of a given organism).

Therefore, if feedback can be performed by a non-*sui generis* computational system, then *sui generis* computation is not constitutive of neural processing. On the contrary, if feedback control cannot be performed by a system that is not a *sui generis* computer, this feature will be also (indirectly) constitutive of neural processing. As we have seen, Piccinini and Bahar (2012) argue that the nervous system is *actually* neither a digital nor an analog computer. Despite this fact, the neural processing *can* be realized in this kind of computing system; *what we call "neural process" can be performed by a digital or analog computer. Sui generis* computation is not constitutive of neural processing.

To determine whether the neural processing can be realized by a digital

or an analog computer, we first have to decide whether feedback control can be performed by a digital or an analog computer. We have seen that neural feedback control is the function of controlling the behavior of an organism in response to stimuli coming from states of the organism's body and environment that are an effect of the feedback system itself. So, for a nervous system to be realized by a digital or an analog computer, digits or strings of digits and continuous variables must be able to cause states in an organism in response to the stimuli coming from states of the organism's body or environment caused by strings of digits or continuous variables. Of course, not every feedback control system that processes digits or continuous variables will be a nervous system. It is also necessary for the system to form part of an organism.

We have seen that negative feedback control, even in its more narrow characterization (the one that included the properties of equifinality and compensation) did not require that its variables be non-continuous or non-discrete. The only structural restriction that these requisites impose on feedback vehicles is that they must be variables capable of going back to an equilibrium value from an arbitrarily large set of non-equilibrium values and that the process of a variable of going back below a threshold depends on the increase of the values of other variables. Some possible changes in the value of the variables and some specific correlation between those changes are the only conditions required. But given these restrictions, it is perfectly possible that these variables be discrete or continuous as digital or analog computation requires. Therefore, feedback control does not imply *sui generis* computation. Any computational vehicle, including digital and analog vehicles, can perform negative feedback control in a given organism. A form of *sui generis* computationalism, the thesis that *sui generis computation* is constitutive of neural processing, is false.

3.2. Constitutive Generic Computation

Piccinini and Bahar (2012) arrived at the conclusion that the nervous system is *actually* a *sui generis* computer, arguing that the nervous system is neither a digital nor an analog computer and that the nervous system is a computer in the generic sense. In section 3.1, I argued that *sui generis* computation is not constitutive of neural processing. It can be realized by an

analog or a digital computer. In this section, I will argue that computation in the generic sense is not constitutive of neural processing either. First, let's briefly consider Piccinini and Bahar's arguments for *actual* generic computationalism.

The first argument is called the 'argument from the functional organization of the nervous system.' The idea is that since (i) neural processes are defined over medium independent vehicles and (ii) the processing of medium independent vehicles constitutes computation in the generic sense, then it follows that (iii) neural processes are computations in the generic sense. Premise (ii) is the result of Piccinini and Scarantino's elucidation of generic computation. So, Piccinini and Bahar only need to argue for premise (i). They do so by pointing out that current evidence indicates that the primary vehicles manipulated by neural processes are neuronal spikes (action potentials) and that the functionally relevant aspects of neural processes depend on dynamical aspects of spikes such as spike rates and spike timing. Only these dimensions of variation of their vehicles, not any more specific property, are functionally relevant for neural processes. This means, according to Piccinini and Bahar, that action potentials are medium independent.

I argued above that computational vehicles are not defined by their medium independence, but rather by their restricted functional relevance. So, as premise (ii) is false, from premises (i) and (ii), conclusion (iii) does not follow. The fact that vehicles of neural processing are medium independent does not imply that they are computational. Nevertheless, the evidence that Piccinini and Bahar offer for premise (ii) also supports the thesis that vehicles of neural processing have not only medium independence, but also restricted functional relevance. So, we can argue for generic computation by reformulating premise (i) as the thesis that computation is the manipulation of vehicles of restricted functional relevance, and reformulating premise (ii) as the thesis that actual neural processing consists of the manipulation of vehicles of restricted functional relevance. These two premises establish generic computation.

The second argument is called 'the argument from information processing.' Piccinini and Bahar affirm that since (i) neural processes process semantic information, and (ii) the processing of semantic

information requires generic computation, then it follows that (iii) neural processes are computations in the generic sense. Here, both premises require some argumentation. Regarding premise (i), Piccinini and Bahar maintain that cognition involves the processing of information in the three senses I mentioned earlier. First, mutual information, as a measure of the statistical dependency between a source and a receiver in Shannon's theory (Shannon and Weaver 1949), is often used to quantify the statistical dependence between neural signals and their sources and to estimate the coding efficiency of neural signals (Dayan and Abbott 2001, ch. 4; Baddeley et al. 2000). There is also evidence that shows that neural signals carry natural semantic information. It is experimentally well established that neural variables carry natural semantic information about other internal and external variables (Adrian 1928; Rieke et al. 1999; cf. Garson 2003). Much of contemporary neurophysiology centers on the discovery and manipulation of neural variables that correlate reliably—and often usefully—with variables external to the nervous system. Lastly, Piccinini and Bahar affirm that many theories in cognitive science assume that nervous systems possess and process representations. In their interpretation, this means that the nervous system possesses and processes non-natural semantic information. Specifically, they say it is difficult to explain language processing and other higher cognitive processes without postulating representations.

Regarding premise (ii), Piccinini and Bahar affirm that information, in the three senses mentioned above, is a medium-independent notion:

Processing information means processing vehicles on the basis of the information they carry rather than their specific physical properties. And the information carried by a signal is defined independently of the physical medium that implements the signal. Thus, processing information requires processing vehicles that are defined in a medium-independent way. Thus, information processing entails generic computation. (The converse does not hold because medium-independent vehicles need not carry any information.) (p. 12)

So, as neuronal spikes carry information in the three main senses, and the

vehicles of information are medium independent, Piccinini and Bahar conclude that the nervous system is computational in the generic sense.

Premise (ii), that the processing of semantic information requires computation, is more complex than it may seem. It contains three different claims: (a) that vehicles of semantic information processing are medium independent, (b) that vehicles of generic computation are medium independent, and (c) that this common feature makes informational vehicles computational. As we have seen, Piccinini and Bahar are correct in claiming (a) and (b). The vehicles of information processing and computation are medium independent. Despite the common feature of medium independence, however, (c) is false. There is no implication from information processing to generic computation, because the medium independence of an informational vehicle type is compatible with no restricted functional relevance of some of its tokens. The argument, then, does not work. Nevertheless, as I believe that my reformulation of the first does work, generic computationalism is true.

What about *modal* generic computationalism, the thesis that neural processing cannot be realized in a system that is not computational in the generic sense? I argued above that feedback control is the constitutive feature of the nervous system. I also argued that, as feedback control implies information processing but not computation, and because information processing does not imply computation either, computation is not constitutive of feedback control. Feedback control can be performed by a non-computational system. From these premises, I can conclude that computation in the generic sense is not constitutive of neural processing. Generic computationalism, as a thesis that generic computation is a constitutive feature of neural processing, is false.

Piccinini and Bahar (2012) do not seem interested in this form of computationalism. They do not consider the *possibility* that neural processing can be realized in a system that is not computational in the generic sense or any other more specific sense. As we have seen, part of their two arguments for generic computationalism is conceptual. The first argument depends on the conceptual elucidation of generic computation as the manipulation of medium independent vehicles. The second argument depends on the same thesis and also on the idea that information processing

conceptually implies manipulation of medium independent vehicles. These conceptual relations describe constitutive relations between properties. An information processing system cannot be realized in a system that does not manipulate medium independent vehicles. A generic computational system cannot be realized by a system that does not manipulate medium independent vehicles, nor can medium independent vehicle processing (in Piccinini and Bahar's view) be realized in a system that is not a generic computer. The other relevant premises of the argument are empirical. They are the empirically well-established facts that the nervous system processes medium independent vehicles and that the nervous system processes information. But no argument is offered to the effect that these empirically discovered features are constitutive of the nervous system itself. If the neural processing can be realized by a system that does not manipulate medium independent vehicles, despite the conceptual relations established by Piccinini and Bahar, neural processing could be realizable by a system that is not computational in the generic sense.

The argument I offer rests also on an empirical premise, that the nervous system is in fact a feedback control system, and three conceptual premises, the fact that feedback control conceptually entails information processing and that neither feedback control nor information processing imply computation. But, unlike Piccinini and Bahar's argument, the empirical premise comes together with a philosophical or conceptual insight about the constitutive character of the feature empirically discovered (negative feedback control).

To conclude this section, it is important to notice a peculiarity of the relation between modal generic computationalism and modal *sui generis* computationalism. Modal *sui generis* computationalism can be understood in two ways: first, as the thesis that neural processing cannot be realized in a system that is not a *sui generis* computer. But it also can be understood as the thesis that *if* the neural processing is realized in a computational system, then it is realized in a *sui generis* computer. The falsity of generic computationalism implies the falsity of the first form of *sui generis* computationalism, but not the falsity of the second. If the neural processing can be realized by a system that is not computational at all, then it can be realized in a system that is not a *sui generis* computer. However, even if the

nervous system can be realized in a system that is not computational at all, it could be the case that the only kind of computational system in which it can be realized is *sui generis*. To argue against the second form of *sui generis* computationalism, we need the argument I offered in the previous section, to the effect that the neural processing can be realized by an analog or digital computing system. This is why the argument in this section cannot substitute the one in the previous section. I want to argue against *both* forms of constitutive *sui generis* computationalism, and in order to challenge the second version, the argument in section 3.1 is required.

4. Conclusion

In the first part of this paper, I have developed two main arguments. First, that computation cannot be defined as the manipulation of medium independent vehicles, but rather the manipulation of vehicles that have what I call ‘restricted functional relevance.’ Second, I defended two ideas that show that the line between computation and information is actually wider than the current debate on computationalism would suggest. I have argued against Piccinini and Scarantino (2011) that it is not only the case that computation and information are different concepts, but also that information processing does not imply computation. The main idea was that any given informational vehicle type can be realized by a token that has no restricted functional relevance. I then argued that information processing is more closely related to the nervous system than computation is, because only the former is constitutive of the peculiar function of the system. I showed that even a weak notion of negative feedback control implies information processing, but that even a strong notion of feedback control does not imply computation.

In the second part of the paper, I used these previous results to consider different forms of computationalism. I challenged a form of the kinds of computationalism that Piccinini and Bahar (2012) defended. However, as ‘computationalism’ was understood here in a different sense than that accepted by Piccinini and Bahar, my theses are compatible with theirs. First, I argued against a form of *sui generis* computationalism. I presented reasons to believe that feedback control is constitutive of the nervous

system in a way that information processing and computation are not. Then I showed that, since feedback control is compatible with digital and analog computation, the neural processing can be realized by an analog or a digital computer. *Sui generis* computation is not constitutive of neural processing. Finally, I questioned a form of generic computationalism. As I established that feedback control vehicles can be realized in tokens that are not computational at all, it follows that generic computation is not constitutive of neural processing.

My two main aims were to clarify several conceptual theses that are important to the debate on computationalism and also to offer answers to new questions regarding computation in the nervous system. First, I intended to go deeper into the characterization of computation, the relation between computation and information, and the relation that they both have with the nervous system. Second, I evaluated computationalism as the thesis that generic and *sui generis* computation are *constitutive* of neural processing. I believe that these theses are more deeply philosophical without being any less naturalistic. As we have seen, they depend on considerations that determine a criterion of multiple realizability for the nervous system. This allows for a defense of the theses about neural processing that goes beyond what actual neural processing is. These theses were answers to the question of whether the neural processing can be realized in a system that lacks some properties that actual nervous systems have. Nevertheless, the criterion of multiple realizability demands that we determine, among the properties *empirically* discovered in the nervous system, the ones necessarily required for the realization of neural processes.

References

- Adrian, E. D. 1928. *The Basis of Sensation: The Action of the Sense Organs*. New York: Norton.
- Ashby, W. R. 1960. *Design for a Brain*, Wiley, New York (2nd revised ed.).
- Baddeley, R. Hancock, P., et al. (eds.) 2000. *Information Theory and the Brain*. Cambridge: Cambridge University Press.
- Beckner, M. 1959. *The Biological Way of Thought*, New York: Columbia Univ. Press.
- Bertalanffy, L. von, 1950. *The Theory of Open Systems in Physics and Biology*,

- Science* 111, 23-29,
- Bertalanffy, L. von, 1950. An Outline of General Systems Theory, *Brit. J. Phil. Sci.* 1, 139-64.
- Braithwaite, R. B. 1953. *Scientific Explanation*, Cambridge University Press, London.
- Cannon, W. B. 1932. *The Wisdom of the Body*. New York: Norton.
- Craver, C. F. 2010. Prosthetic Models. *Philosophy of Science* 77(5), 840-851.
- Dayan, P. and L. F. Abbott 2001. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. Cambridge, MA: MIT Press.
- Ermentrout, G. B. and Terman, D. H. 2010. *Mathematical Foundations of Neuroscience*. New York: Springer.
- Garson, J. 2003. The Introduction of Information into Neurobiology. *Philosophy of Science* 70, 926-936.
- Grice, H. P. 1957. Meaning. *The Philosophical Review*, 66(3), 377-388.
- Kacser, H. 1957. Some Physico-Chemical Aspects of Biological Organization, printed as an appendix to C. H. Waddington's *The Strategy of the Genes*, New York: Macmillan, 191-249.
- Kleene, S. C., 1956. Representation of Events in Nerve Nets and Finite Automata. In C. E. Shannon and J. McCarthy (Eds.) *Automata Studies*. Princeton, NJ: Princeton Univ. Press, 3-42.
- Lin, J.-W. and Faber D. S. 2002. Modulation of Synaptic Delay During Synaptic Plasticity. *Trends in Neurosciences* 25(9), 449-455.
- McCulloch, W. S. and Pitts W. H. 1943. A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics* 7, 115-133.
- Mills, J. W. 2008. The Nature of the Extended Analog Computer. *Physica D: Nonlinear Phenomena*, 237(9), 1235-1256.
- Nagel, E., 1961. *The Structure of Science*. New York: Harcourt.
- Piccinini, G. 2007. Computing Mechanisms. *Philosophy of Science* 74(4), 501-526.
- Piccinini, G. 2004. The First Computational Theory of Mind and Brain: A Close Look at McCulloch and Pitts's 'Logical Calculus of Ideas Immanent in Nervous Activity'. *Synthese* 141(2), 175-215.
- Piccinini, G. and Scarantino, A. 2011. Information processing, Computation and Cognition. *Journal of Biological Physics* 37.1, 1-38.
- Piccinini, G. and Bahar, S. 2012. Neural Computation and the Computational Theory of Cognition. *Cognitive Science* 37-3, 453-488.
- Pour-El, M. B. 1974. Abstract Computability and Its Relation to the General Purpose Analog Computer (Some Connections Between Logic, Differential Equations and Analog Computers). *Transactions of the American Mathematical Society* 199, 1-28.
- Rosenblueth, A., Wiener, N., and Bigelow, J., 1941. Behavior, Purpose, and

- Teleology. *Phil. Sci.* 10, 18-24.
- Rosenblueth, A. and Wiener, N., 1950. Purposeful and Non-Purposeful Behavior. *Phil. Sci.* 17, 318-26.
- Rubel, L. A. 1993. The Extended Analog Computer. *Advances in Applied Mathematics* 14(1), 39-50.
- Shannon, C.E. 1948. A mathematical theory of communication. *ATT Tech. J.* 27 (379-423), 623-656.
- Shannon, C. E., and Weaver, W. 1949. *The Mathematical Theory of Communication*. Illinois: Univ of Illinois Press.
- Sommerhoff, G., 1950. *Analytical Biology*. London: Oxford Univ. Press.
- Wiener, N., 1961. *Cybernetics* (2nd revised ed.). Cambridge, Mass.: M.I.T. Press.
- Wimsatt, W. 1970. Some Problems with the Concept of Feedback. *PSA-1970*, pp. 241-256.