

University of Windsor

Scholarship at UWindor

Philosophy Publications

Department of Philosophy

1996

Tensor Products and Split-Level Architecture: Foundational Issues in the Classicism-Connectionism Debate

Marcello Guarini
University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/philosophypub>



Part of the [Philosophy Commons](#)

Recommended Citation

Guarini, Marcello. (1996). Tensor Products and Split-Level Architecture: Foundational Issues in the Classicism-Connectionism Debate. *Philosophy of Science*, 63 (3 (Supplement)), 239-247.
<https://scholar.uwindsor.ca/philosophypub/17>

This Article is brought to you for free and open access by the Department of Philosophy at Scholarship at UWindor. It has been accepted for inclusion in Philosophy Publications by an authorized administrator of Scholarship at UWindor. For more information, please contact scholarship@uwindsor.ca.



CHICAGO JOURNALS



Tensor Products and Split-Level Architecture: Foundational Issues in the Classicism-Connectionism Debate

Author(s): Marcello Guarini

Reviewed work(s):

Source: *Philosophy of Science*, Vol. 63, Supplement. Proceedings of the 1996 Biennial Meetings of the Philosophy of Science Association. Part I: Contributed Papers (Sep., 1996), pp. S239-S247

Published by: [The University of Chicago Press](#) on behalf of the [Philosophy of Science Association](#)

Stable URL: <http://www.jstor.org/stable/188532>

Accessed: 31/10/2012 16:44

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at

<http://www.jstor.org/page/info/about/policies/terms.jsp>

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.



The University of Chicago Press and Philosophy of Science Association are collaborating with JSTOR to digitize, preserve and extend access to *Philosophy of Science*.

<http://www.jstor.org>

**TENSOR PRODUCTS AND SPLIT-LEVEL ARCHITECTURE:
FOUNDATIONAL ISSUES IN THE
CLASSICISM-CONNECTIONISM DEBATE**

MARCELLO GUARINI†‡

University of Western Ontario

This paper responds to criticisms levelled by Fodor, Pylyshyn, and McLaughlin against connectionism. Specifically, I will rebut the charge that connectionists cannot account for representational systematicity without implementing a classical architecture. This will be accomplished by drawing on Paul Smolensky's Tensor Product model of representation and on his insights about split-level architectures.

1. The 1980s saw an explosion in the amount of connectionist (or PDP) research being done. Some found the results so exciting they talked of the emergence of a new paradigm in cognitive science. A Kuhnian revolution was on the horizon; connectionists of the world, UNITE! Well, Fodor and Pylyshyn's (F&P) 1988 paper "Connectionism and Cognitive Architecture: A Critical Analysis" was something of a cold shower for connectionists. It was an attempt to show that connectionist architectures were, at best, mere implementations of the classical view of cognition, or, at worst, woefully inadequate as models of real cognition. Four arguments were marshalled to bolster that dilemma: the productivity of thought, the compositionality of thought, representational systematicity, and inferential systematicity. Paul Smolensky (and others) began the task of responding to these arguments, and Fodor and McLaughlin (F&M) replied in 1990 with "Connectionism and the Problem of Systematicity: Why Smolensky's Solution Doesn't Work." I will defend Smolensky against some of F&M's criticisms by showing that he has begun the task of responding to F&P's criticisms. While F&P have made some valid arguments against certain forms of connectionism, the focus of this paper will be on where they have overdrawn their conclusions. So while my tone will be generally critical, I should not be interpreted as disagreeing with all of F&P's criticisms. Indeed, a good portion of this paper is a clarificatory recasting of Smolensky's work, a recasting made necessary by the force of F&M's objections.

The layout of the paper is as follows. In part two, I summarize what F&P take to be the essential features of classical architecture and present Smolensky's tensor product proposal for composed and systematic representations, defending the latter against F&M's charge that it does not account for systematicity in a non-classical manner. In part three, I discuss Smolensky's claim that the split-level nature of some connectionist architectures makes them importantly different from classical ones. F&M have some important concerns about split-level architectures, but I think they can be answered without commitment to the strong form of re-

†I thank Ausonio Marras for comments on an earlier draft of this paper, and the Social Sciences and Humanities Research Council of Canada for financial support, award 752-94-1127.

‡Department of Philosophy, University of Western Ontario, London, Ontario, N6A 3K7.

ductionism they fear. I will close with some remarks about the implications of connectionism for Intentional Realism and the Representational Theory of Mind.

2. Let us begin with what F&P take to be some of the essential commitments of classicism. They claim that since classical models are committed to mental representations having combinatorial syntax and semantics, such models require the Language of Thought (LOT) hypothesis, which is committed to the following theses:

- (a) there is a distinction between structurally atomic and structurally molecular representations; (b) structurally molecular representations have syntactic constituents that are themselves either structurally molecular or structurally atomic; and (c) the semantic content of a (molecular) representation is a function of the semantic contents of its parts, together with its constituent structure. (1988, 13)

Another commitment of the language of thought is that (d) when a syntactically (and semantically) molecular representation is tokened, so too are its syntactic (and semantic) constituents (1988, 13). For example, the sentence “Jack loves Jill” cannot be tokened without tokening the words “Jack,” “loves,” and “Jill.” This might seem like a trivial observation, but it will loom large later on, and we will see that it is worth making this commitment explicit.

According to F&P, to say that thought is systematic is to say that “the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others” (1988, 35). For example, creatures that can think “Jack loves Jill” tend to have the ability to think “Jill loves Jack” and “Jack loves Jack” and “Jill loves Jill.” To see how a classicist could account for systematicity, let us assume that sentences of the form **NP Vt NP** (where **NP** is a noun phrase and **Vt** is a transitive verb) are well formed formulae in our language of thought. If we think in a language, and if formulae having the form **NP Vt NP** are well formed in that language, then we can use all the mental vocabulary available to us to think thoughts of that form. If “Jack” and “Jill” are **NPs** and “loves” is a **Vt**, then we have the capacity to think all the thoughts involving the given vocabulary and grammatical form. F&P have some convincing arguments against the ability of some connectionist models to account for systematicity, but I will not examine them here since I am willing to concede that many connectionist models cannot be used to explain systematicity. What I will challenge is the claim that connectionist models can only account for systematicity by implementing a classical model.

A lower level account of cognition implements a higher level account if and only if the higher level is a complete, precise, algorithmic account of the evolution of the system in question. For example, Machine Language implements Fortran, Pascal, BASIC, and a number of other higher level computer languages, but each of these higher level languages can be used to give a complete, precise, algorithmic account of the evolution of the systems driven by them. This means that higher and lower level accounts are isomorphic. On the classical view, connectionism will yield nothing more than some sort of machine language appropriate for driving the massively parallel pieces of hardware we call brains. Classical models are alleged to be complete, precise, algorithmic accounts of cognition, and all connectionism will yield is an implementation theory for classical models.

What if a neural net (i) had representational states, (ii) was capable of some sort of structure sensitive processing, (iii) could account for systematicity in virtue of that sort of processing, and (iv) did not token the components of “complex” rep-

representations while tokening those representations? I will argue that such a neural net does not implement classical architecture.

Tensor products allow us to construct non-classical accounts of representation which can be implemented in a straightforward manner. To tell this story, we need to get two mathematical concepts under our belts—vector addition and tensor products. For any two n -element vectors \mathbf{v} and \mathbf{w} , their sum ($\mathbf{v} + \mathbf{w}$) is the n -element vector whose first element is the sum of the first element of each vector, and whose second element is the sum of the second element of each vector, and so on. So much for vector addition. For any two vectors \mathbf{v} and \mathbf{w} having n and m elements respectively, their tensor product will be the $(n \times m)$ -element vector whose elements are generated by the pairwise multiplication of each element of \mathbf{v} with each element of \mathbf{w} .

The basic idea of tensor product (or TP) representations is this: use tensor products to create role-invariant representations, and use vector addition on tensor products to build up complex representations out of the simple TP representations. Say that \mathbf{F} is the set of n -element vectors whose members we call ‘fillers,’ and \mathbf{R} is the set of m -element vectors whose members are the ‘roles’ those fillers can have. The representation for a filler playing a particular role will be the $(n \times m)$ -element vector arrived at by taking the tensor product of a vector from \mathbf{F} and a vector from \mathbf{R} . Tensor products can be added together to create more complex representations.

An example is in order. Say we have this state of affairs—Jack loves Jill. The vectors \mathbf{f}_1 , \mathbf{f}_2 , and \mathbf{f}_3 respectively represent Jack, loving, and Jill; the vectors \mathbf{r}_1 , \mathbf{r}_2 , and \mathbf{r}_3 respectively represent the roles of subject, verb, and object. We are now in position to say that

$\mathbf{f}_1 \otimes \mathbf{r}_1$ represents the representation of Jack in the role of subject,
 $\mathbf{f}_2 \otimes \mathbf{r}_2$ represents the representation of loving in the role of verb, and
 $\mathbf{f}_3 \otimes \mathbf{r}_3$ represents the representation of Jill in the role of object.

$(\mathbf{f}_1 \otimes \mathbf{r}_1) + (\mathbf{f}_2 \otimes \mathbf{r}_2) + (\mathbf{f}_3 \otimes \mathbf{r}_3)$ represents the fact that Jack loves Jill.

We need not restrict ourselves to the sorts of fillers and roles present in this example. Vectors representing the representation of *loving Jill* can be fillers, and grammatical categories such as verb phrase can specify the role. Indeed, all kinds of different structures can function as fillers, and many different types of categories can be used to specify roles.

We can implement tensor product representations in the network depicted in Figure 1. The triangles are multiplicative junctions which multiply the input vectors and add their product to the activation level of the binding units (whose default values we will say are all zero). Binding units are said to bind fillers to roles since they store the product of a filler in a particular role and even the sum of different fillers in different roles. For example, if \mathbf{f}_1 and \mathbf{r}_1 are sent into the net, their product will be passed on to the binding units and a representation for the representation of John in role of subject will be tokened in the binding units. If \mathbf{f}_2 and \mathbf{r}_2 also are fed into the net, then their product will be added to the first product; and if \mathbf{f}_3 and \mathbf{r}_3 are then added into the net, then their product will be added to the sum of the first two products, yielding the vector (which consists of the activation levels in the binding units) which represents that fact that Jack loves Jill. This sort of net can also unbind representations. Unbinding takes place when, say, the query vector representing Jack is given as input while the vector representing the fact that Jack loves Jill is tokened in the binding units, producing the vector for the subject role in the role units. In other words, the network can be “asked” what role the rep-

classical models. While there is a precise mathematical sense in which we can speak of the components of complex TP representations, these representations do not have constituent structure. Given the way I am using the term *constituent*, the component vectors of complex tensor product representations are not constituent vectors. Since it would be useful to have a term to mark the key difference between classical and tensor product representations, I have decided to limit the use of the term *constituent* to cases where constituents are tokened when complex representations are tokened—to classical models and not tensor product models. What we need to preserve is some way of talking about complex representations being built out of simpler ones, but talk of component vectors being multiplied and added is all we need to account for simpler and more complex vector representations.

Above, I pointed out that the constituent structure of classical representations would loom large. Now, we are in position to see why. Classical architectures require mental representations to have constituent structure, and the manipulation or processing of these representations will require the manipulation of their constituents; TP representations have no constituents, so the principles of processing governing these representations will be importantly different from classical principles. This is not to say that composed tensor products will never be decomposed into components; the point is that the causal processes which carry that task out are not isomorphic to the causal processes in classical systems which decompose a sentence into its constituents. If there is no isomorphism between the manipulation of representations with constituents and representations without, then the latter cannot be an implementation of the former. Since an essential feature of all classical architectures is that the story they tell at the syntactic level of description will support constituent structure, and since Smolensky's tensor product story does not support constituent structure, it cannot be a mere implementation of some classical story.

Okay, so the TP story is not an implementation of some classical story, but can it account for systematicity and structure sensitive processing in general? Well, it appears to have a good start on accounting for representational systematicity. The net described above gives us a straightforward way of accounting for systematicity. If we have a neural net that is wired up in that manner, and if we have the simple filler and role vectors in our head, then the network can process them in a systematic manner. Given the vectors for Jack, loving, and Jill, and given the vectors for subject, verb, and object, it is perfectly clear that the net can construct representations for Jack loving Jill, Jill loving Jack, Jill loving Jill, and Jack loving Jack. Moreover, the network can even decompose complex representations. Recall, the network can reply with the vector representing Jack when it was asked for the subject of the representation of the fact that Jack loves Jill. In other words, the network can do processing which is sensitive to the (superposed-TP) structure of the representation for Jack loving Jill even if that representation does not have constituent structure. We have a net that gives us an account of representational systematicity and of rudimentary structure sensitive processing, and it does this by using the superposed TP structure of vectors and not the constituent structure of classical representations.

Now, we can start to look at F&M's objections. One objection they have is that there is no unique way to decompose a vector. It is clear that "Jack loves Jill" can be decomposed into "Jack," "loves," and "Jill," but how do we know that, say, " $\langle 1,0,1,0,1,0,1,0,1,0,1 \rangle$ " is to be decomposed into " $\langle 1,1,0 \rangle$," " $\langle 0,0,1 \rangle$," and " $\langle 1,1,1 \rangle$ " (not to mention the role vectors)? Looking at the network displayed in figure 1, it is tempting to answer as follows: that network is wired up in such a way that it can carry out the composition and decomposition of vectors.

That might seem like a simple-minded response, but even the classicist is committed to a similar move. After all, how does a classical system know to decompose “Jack loves Jill” into “Jack,” “loves,” and “Jill” and not into, say, “Ja,” “cklov,” and “esJill”? The answer will be something like:

well, silly, a classical system is wired up in such a way that it knows how to interpret spaces: a classical mechanism for the decomposition of sentences uses its knowledge of spaces (or their LOT equivalent) and what they represent in order to decompose sentences.

If that kind of explanation is acceptable, then it should not seem so strange if the defender of TP representation says:

well, silly, a tensor product net is wired up in such a way that it knows the principles of vector addition and (tensor) multiplication: a tensor product mechanism for the decomposition of sentences makes use of this knowledge when it decomposes complex tensor product representations.

What F&M have to do to discredit such a move is to show that the principles of vector decomposition just alluded to do not exist. They have not done so; moreover, Smolensky has begun the work of laying out such principles and specifying their limits (to which much of his 1990 is devoted).

It might be objected (in the spirit of F&M) that while you can wire up a net to uniquely decompose vectors, unique decomposition is not built into connectionist architecture. I anticipate this objection, since it is similar to F&M’s objection to Smolensky’s tensor product account of systematicity:

No doubt it is possible for Smolensky to wire a network so that it supports a vector that represents aRb if and only if it supports a vector that represents bRa . . . The trouble is that, although the architecture permits this, it equally permits Smolensky to wire a network so that it supports a vector that represents aRb if and only if it supports a vector that represents zSq ; or, for that matter, if and only if it supports a vector that represents *The Last of The Mohicans*. (1990, 202)

This view is repeated a little further on:

So then: it is *built into* the Classical picture that you can’t think aRb unless you able to think bRa , but the Connectionist picture is *neutral* on whether you can think aRb even if you can’t think bRa . (1990, 203).

The response that I propose for these quoted objections will allow us to respond to the objection I anticipated at the beginning of this paragraph. I think the problem is that “connectionist architecture” is poorly defined. It appears that any architecture which is non-classical and has a massively parallel implementation is connectionist. Of course connectionist architecture will be constitutionally neutral on systematicity if it is defined in the preceding manner, but nothing follows. Tensor Product architecture (as developed by Smolensky) is not neutral on systematicity. The network shown in figure 1 is only one way of implementing a Smolenskian TP architecture; there may be other ways. My point is that it is essential to Smolenskian TP architecture that it support systematicity. Given representations for a , R , and b as fillers, and given a set of roles, tensor products of roles and fillers, and vector addition on those products guarantee that we will be able to represent aRb and its systematic variants. If our architecture says that the way to represent aRb is to add together tensor products of roles and fillers (as described above), then the architecture guarantees that we can represent the systematic var-

variants of aRb (since the role and filler vectors change but the procedures for manipulating them—tensor multiplication and vector addition—are indifferent to that change). The requirement for massively parallel implementation notwithstanding, connectionist architectures have no essential, defining features. Rather than attacking something so poorly defined, F&M&P should be attacking the specific architectures that are loosely referred to as connectionist. F&P and F&M do this when they criticize microfeature architectures and weakly distributed architectures, but when it comes to criticising TP architecture, connectionist architecture in general becomes the target. If we understand the TP approach to representation as an architecture, then F&M cannot argue that it is inferior as a result of not supporting systematicity in an essential manner, because it does.

The sort of argument that I have been giving can be marshalled to rebut F&M's concern that there is no unique way to decompose a vector. In general, there is no unique way to carry out a decomposition. In a specific TP architecture, we know the fillers and roles that went into producing a composed tensor product representation, and that knowledge together with the knowledge of how vectors are added and multiplied allows us to decompose it into its roles and fillers. It is unfair to compare classical decomposition ("Jack loves Jill" into "Jack," "loves," and "Jill") to vector decomposition in general and to argue that the former is superior to the latter since the latter does not support unique decomposition. Putting the task of vector decomposition in the context of a TP architecture makes it very unsurprising that decomposition yields role and filler vectors. That is what the goal of decomposition is in such an architecture, and this should not be surprising since this architecture was designed to account for the component structure of complex representations and for systematicity. Similarly, words (or phrases) are the goals of decomposition in classical models, and in that context it just seems wrong-headed to ask why "Jack loves Jill" is not decomposed into "Ja," "cklovesJ," and "ill." The goals of decomposition are architecture relative. If a specific architecture does not specify the goals of decomposition and the principles for achieving the goals, then we might run into some difficulties. However, we can claim that role and filler vectors are the goals of decomposition in TP architecture, and we can try to formulate principles of decomposition which yield the desired goals. F&M have not shown that such architecture-dependent principles of decomposition cannot exist.

3. As promised, I will now get into the issue of split-level architecture. Smolensky argues that non-linear connectionist architectures using distributed representations have a fundamentally split-level structure. The complete, precise, algorithmic account of the evolution of a cognitive system is given at the level of individual units, whereas the account given at the representational level (where representations are patterns of activation across units) is not a complete, precise, algorithmic account of cognition. This makes distributed, representational, connectionist architectures different from classical architecture since the latter gives a complete, precise, algorithmic account of cognition at the representational level. This point can be appreciated by analogy with computer languages. A lower level language implements a higher level language if and only if the higher level language provides a complete account of cognition. This is to say that lower level accounts can be put into isomorphism with higher level accounts. While that isomorphism between lower and higher levels holds for classical architectures, it does not hold for split-level architectures. Hence the expression *split-level*.

F&M respond to the above (and other) points by remarking that we can get more precision by descending to lower levels of explanation without claiming that the higher levels are illegitimate (1990, 203–204). Even if chemical explanations

turn out not to be reducible to quantum mechanical ones, it does not follow that the former are illegitimate; if psychological (or representational) explanations cannot be reduced to unit or neuron sensitive operations, representational explanations are not automatically discredited. No complaints so far, however, F&M show no appreciation for Smolensky's claim that split-level architectures are importantly different from classical ones. F&M's talk of descending from unit-sensitive operations to molecule-sensitive operations to quark-sensitive operations might give the impression that the lower level of split-level architectures is merely a level which takes into consideration hardware concerns: it is an implementation level concern, not a matter of cognitive architecture. It is worth emphasising that Smolensky is *not* making the claim that we get more precision as a result of factoring hardware concerns into the dynamical equations governing a system. To see this, imagine that a classical architecture is implemented in a neural net. If there are no unexpected physical problems with the neurons, then the representation-level description will give us a complete account of the evolution of the net. If we implement a split-level architecture into the same physically flawless neural net, the complete account of the evolution of the net can only be given at the level of individual units—at the sub-representational level. Holding physical concerns constant, classical architecture gives us completeness and precision at the representational level, but split level architectures do not. The importance of this point will be brought out shortly.

4. Part three of this paper can be taken as a defense of the distinctness of split-level (connectionist) architectures from classical architecture. Even if this defense fails, part two shows that there are other grounds for regarding some connectionist architectures (namely, Tensor Product architecture) as distinct from classical architecture. The reason for this is that classical architecture requires constituent structure and TP architecture does not. If TP architecture (or something like it) turns out to be the one-and-only true cognitive architecture, then the LOT hypothesis is false. The reason is that LOT requires representations to have constituent structure. However, Intentional Realism (IR) and the Representational Theory of Mind (RTM) are not necessarily affected by the truth of some non-classical architecture. That a vector with non-sentential properties expresses a proposition shows that the RTM can be true while LOT is false. If the one-and-only true cognitive architecture is split level, then IR and RTM are not obviously affected. On the classical approach, if we hold physical variables constant, then a representation-level story will give a complete account of cognition. If some split-level architecture turns out to be true, then classicism is importantly incomplete, but it does not follow (without independent considerations) that IR and RTM are wrong. The truth of a split-level architecture of the sort we have considered entails that the complete account of cognition is given at the sub-representational level, but Fodor never defined IR and RTM in such a way that they entailed the completeness of representation-level descriptions. It is possible to argue from within a split-level architecture that intentional generalizations are roughly correct, that having a thought is to be in relation to a vector (as opposed to a sentence) which expresses a proposition, and that the physical analogue of this meaning bearing vector has causal efficacy. This would be enough to make IR and RTM true even if there is a lower level story which is more precise, or so one might be inclined to argue. The reason this position is not obviously implausible is that inferring a lack of causal efficacy on the basis of there being a lower level story which is more complete is dubious, as we saw above. The falsity of IR and RTM does not follow (without further argument) from the truth of split-level architecture, but the in-

completeness of classicism does. If a suitably powerful type of TP architecture turned out to be true, then classicism would be false.

REFERENCES

- Fodor, J. and McLaughlin, B. P. (1990), "Connectionism and the Problem of Systematicity: Why Smolensky's Solution Doesn't Work", *Cognition* 35: 183–204.
- Fodor, J. and Pylyshyn, Z. (1988), "Connectionism and Cognitive Architecture: A Critical Analysis", *Cognition* 28: 3–71.
- Horgan, T. and Tienson, J. (1992), "Structured Representations in Connectionist Systems?", in S. Davis (ed.), *Connectionism: Theory and Practice*. Oxford: Oxford University Press, pp. 195–228.
- Smolensky, P. (1990), "Tensor Product Variable Binding and the Representation of Symbolic Structures in Connectionist Systems", *Artificial Intelligence* 46: 161–216.
- . (1991), "Connectionism, Constituency, and the Language of Thought", in B. Loewer and G. Rey (eds.), *Meaning in Mind: Fodor and his Critics*. Cambridge MA: Blackwell, pp. 201–227.