The centrality of instantiations

John A. Barnden

Extracted, with typographical edits and addition of some references, from *Behavioral and Brain Sciences 10* (3): 437–438 (1987).

(It’s a commentary on the target article by Michael Arbib, “Levels of modeling of mechanisms of visually guided behavior”, in the same issue of the journal, pp. 407–465.)

I applaud Arbib's incorporation of multiple, coexistent instantiations of schemas as a central feature of his framework. Not enough attention has been given in neural net and connectionist research to multiple instantiations of schemas, frames, situations, scripts, and so forth. In fact, insofar as Arbib's target article is about the general nature of a schema framework, I see its main substance as lying in the notion of multiple instantiations. If Arbib's general view of schemas did not encompass multiple instantiations, then one would presumably view schemas themselves as relatively permanent, interacting, concurrent processes. There is nothing particularly novel about the individual abstract schema programs Arbib proposes, and the general idea that information processing in the brain is based on interacting processes is hardly new. This, compounded with Arbib's understandable desire for a general framework rather than a particular specification language for schemas (see sec. 2, para. 1), means that without multiple instantiations we would not be left with very much in the way of a substantive proposal about abstract schemas in general. On the other hand, the inclusion of multiple instantiations presents deep, revealing, and approachable problems when the task of implementing the schema level in neural circuitry is considered.

I do not mean that multiple instantiations exhaust the substance of Arbib's target article in all its aspects. First, I am not impugning the intrinsic interest of the neural-level modeling efforts he reports. Second, the notion of cooperative computation would be important and interesting even in the absence of multiple schema instantiations. There are problems to be tackled in determining the nature of communication among cooperating agents and the means by which agents absorb incoming information. These are pressing issues in computer science, artificial intelligence, and cognitive science (including connectionism). In the Section 5 depth-finder example, the two cooperating subsystems (one disparity based, one accommodation based) can, I take it, be viewed as unique instantiations of two different schemas. This uniqueness does not mitigate the worth of the model or the schema notion underlying it.

I also find Arbib's plea for an intermediate level between behavior and neurons a refreshing corrective to the lack of such a level in much neural-net and connectionist work. Quite apart from the point that intermediate levels are a heuristic aid in managing a large research effort, they can also free one to discover useful reductions of high-level notions to low-level ones that could otherwise escape one's attention - through being, perhaps, too complex or deviant to conceive of unitarily. [See endnote 1.]

Arbib is right to emphasize multiple schema instantiations, but I wish that he had pursued the matter further. The depth-finder example of Section 5 is prominent in the paper, yet it does not involve multiple instantiations. The other examples, in the main text and Appendix B, do not involve detailed attention to multiple instantiations, and there is no consideration of how multiple instantiations would be neurally realized. As I shall now discuss, the multiple-instantiation issue has relationships to longstanding issues in brain theory and connectionism, and poses interesting problems for these fields.

One old problem in connectionist research is that of avoiding "cross-talk" between different pieces of information. For example, if one supposes that a connectionistically implemented agent is simultaneously entertaining the ideas that John loves Mary and Bill loves Sally, one has to take a certain amount of care to ensure that the agent's internal state is not also one that would obtain if the propositions were instead that John loves Sally and Bill loves Mary [and see endnote 2]. A standard approach to the issue is based on recruiting neurons or neural assemblies to represent the particular instances of a situation class (e.g., loving) as well as having a neural assembly standing for the situation class itself (see Hinton 1981). Recruitment is also advantageous in accounting for the ability of an agent to entertain any novel, short-term, complex propositions (or data structures) in the first place. The notion of temporary, short-term recruitment in turn raises such questions as, for example: How is it managed? Does it rely on synaptic-weight change or on some other mechanism? How can it be made fast enough? How economically can inferential and other information-processing mechanisms respond to neural structures involving recruited - and therefore in a sense unpredictable - assemblies? How are recruited assemblies demobilized? - a nontrivial question at least in the case of "distributed" connectionist systems in which a given assembly can share many neurons with other assemblies. The report of a recent workshop on connectionism (McClelland et al. 1986) identifies cross-talk and the more general issue of accounting for possibly novel, complex temporary data structures as being of major concern. The relevance to the multiple-instantiation issue is clear - simultaneous presence of several propositions about loving is similar to, or even an example of, the simultaneous presence of several instantiations of an Arbib schema.

Suppose for definiteness that a schema S is neurally realized as some particular neural net N. What then does it mean to say that an instantiation of S is present? If only one instantiation at a time were ever allowed, an instantiation's existence could simply be a matter of certain state parameters of N having certain values - e.g., certain neurons having certain firing rates. But in the multiple, coexisting instantiation case, things are not so simple - e.g., we cannot suppose that a neuron can simultaneously be firing at several different rates. We can ask general questions such as: Can we produce a theory in which different instantiations are somehow superimposed states of N itself? Or are the different instantiations different copies of N in some sense (these copies being dynamically recruited, or perhaps permanently existing but only intermittently active)? Or do the different instantiations involve neural networks that are not copies of N but are temporary holders of different sets of parameter values for S and different "program counters" for S (so that the networks might be copies of only a part of N)? If instantiations involve neuronal recruitment, what are the answers to the above questions about recruitment? How are the answers to all these questions affected by a consideration of the number of instantiations of a given schema that can be simultaneously present?

This discussion of instantiations can be modified to account for the possibility that it is only instantiations, and not schemas themselves, that are identifiable with neural circuits. A schema could, for instance, be construed merely as a propensity to create or activate neural circuits of a certain form that act as instantiations.

A special problem concerning data communication among computing agents arises for a brain-oriented framework like Arbib's. There are several different types of neural encoding that might be envisaged for the data. For example, on one channel of communication the information might take a retinotopic form; on another, it might be encoded in the firing rates of some neurons with no special organization. I appreciate that at the schema level Arbib may not wish to take such factors into account because they are implementation details (although he is sensitive to them, as point 2(b) in section 6 shows). At the same time, I suspect that such factors should be allowed to affect abstract schema formalisms. For example, a particular type of encoding (perhaps by single-neuron firing rate) may have a large inherent imprecision, so that only data channels that could tolerate that imprecision should be so encoded, whereas other types of encoding (e.g., fine-grained retinotopic encoding of position information) may involve much less imprecision. It might therefore be advantageous to specify, in a schema formalism itself, the degrees of precision that various channels (ports) require. This would (a) provide a guard against uncritically allowing schemas to transfer values on low-precision channels to high-precision ones. More generally, it would (b) help to ensure that the behavioral part of a schema does not involve excessive hidden conversion between different sorts of encoding. It would also (c) usefully constrain the types of encoding allowed in implementing the schema system at hand in neural circuitry. Points (b) and (c) suggest that a schema formalism could even allow particular encoding techniques to be specified or suggested for channels.

There is another enhancement to the schema framework that I would like to propose and that I suspect would be congenial to Arbib. Appendices A and B imply that the behavioral part of a schema is to be specified by means of a fairly conventional analogue to current programming languages (although the imprecise description of schemas in the main text leaves the door open to other sorts of behavior specification). The suggested enhancement is to allow the behavior to be specified instead by means of a mathematical description of an input-output function linking input ports to output ports, or, more generally, of a relation (in the set-theoretic sense) linking the ports. The introduction of a program to compute the function or relation could then be part of the task of implementing the schema in lower-level terms; and it may not even be appropriate to think of using a program (as opposed to, say, a connectionist subsystem with no convenient abstract algorithmic characterization) on the implementation route. I would in fact favor a hybrid approach in which both programs and mathematical descriptions are available as behavior-specification tools.

NOTES

1. In my own research on connectionist models for complex short- term inferential information processing I have eschewed the typical idea of thinking of abstract connectivity among pieces of information as being mapped in any direct way onto hardware connectivity among units in a network. Instead, the mapping appeals to an intermediate level in which there is a notion of relative position of data items in certain representational media. This relative position at the intermediate level manifests itself in terms of hardware connectivity in a complex way (see Barnden 1985; 1986).

2. In some contexts, however, one might want to claim that cross-talk effects accurately reflect human information processing. This approach is taken in a limited visual-processing context by Hinton and Lang (1985).

REFERENCES to my articles [for others' articles, please see the target paper]

Barnden, J. A. (1985) Diagrammatic short-term information processing by neural mechanisms. *Cognition and Brain Theory* 7:285-328.

Barnden, J. A. (1986) Complex cognitive information processing: A computational architecture with a connectionist implementation. Technical Report 211, Computer Science Department, Indiana University, Bloomington. *See also/instead the following references::*

Barnden, J.A. (1994). Complex symbol-processing in Conposit, a transiently localist connectionist architecture. In R. Sun & L. Bookman (Eds), *Computational Architectures for Integrating Neural and Symbolic Processes,* pp.21-68. Kluwer.

Barnden, J.A. (1995). High-level reasoning, computational challenges for connectionism, and the Conposit solution. *Applied Intelligence, 5*(2), pp.103-135.