Grounding Analog Computers

Bruce J. MacLennan
Computer Science Department
University of Tennessee
Knoxville, TN 37996
Email: maclennan@cs.utk.edu

1.0 Is Cognition Discrete or Continuous?

1.1 Although I hate to haggle over words, Harnad's use of 'analog' confuses a number of issues. The problem begins with the phrase 'analog world' in the title, which does not correspond to any technical or nontechnical usage of 'analog' with which I'm familiar. Although I don't know precisely what he means by 'analog', it is clearly related to the distinction between analog and digital computers, so I'll consider that first.

1.1.1 In traditional terminology, analog computers represent variables by continuously-varying quantities, whereas digital computers represent them by discretely-varying quantities (typically, voltages, currents, charges, etc. in both cases). Thus the difference between analog and digital computation lies in a distinction between the continuous and the discrete, but it is not the precise mathematical distinction. What matters is the behavior of the system at the relevant level of analysis. For example, in an analog computer we treat charge as though it varies continuously, although we know it's quantized (electron charges). Conversely, in a digital computer we imagine we have two-state devices, although we know that the state must vary continuously from one extreme state to the other (voltage cannot change discontinuously). The mathematical distinction between discrete and continuous is absolute, but irrelevant to most physical systems.

1.1.2 Many complex systems are discrete at some levels of analysis and continuous at others. The key questions are: (1) What level of analysis is relevant to the problem at hand? (2) Is the system approximately discrete or approximately continuous (or neither) at that level? One conclusion we can draw is that it can't matter whether an analog computer system (such as a neural net) is "really" being simulated by a digital computer, or for that matter whether a digital computer is "really" being simulated by an analog computer. It doesn't matter what's going on below the level of relevant analysis. So also in the question of whether cognition is more discrete or more continuous, which I take to be the main issue in the symbolic/connectionist debate. This is a significant empirical question, and the importance of connectionism is that it has tipped the scales in favor of the continuous.

1.2 Having considered the differences between analog and digital computers, I'll now consider their similarities, which I think are greater than Harnad admits.

1.2.1 First, both digital and analog computers provide *state spaces*, which can be used to represent aspects of the problem. In digital computers the set of states is (approximately) discrete, e.g., most of the time the devices are in one of two states (i.e., 0 and 1). On the other hand, in analog computers the set of states is (approximately) continuous, e.g., in going from 0 to 1 it seems to pass through all intermediate values. In both cases the physical quantities controlled by the computer (voltages, charges, etc.) correspond to quantities or qualities in the problem being solved (e.g., velocities, masses, decisions, colors).

1.2.2 Both digital and analog computers allow the programmer to control the trajectory of the computer's state through the state space. In digital computers, *difference equations* describe how the state changes discretely in time, and programs are just generalized (numerical or nonnumerical) difference equations (MacLennan 1989, 1990a, pp. 81, 193). On the other hand, in analog computers, *differential equations* describe how the state changes continuously in time. In both cases the actual physical quantities controlled by the computer are irrelevant; all that matters are their "formal" properties (as expressed in the difference

or differential equations). Therefore, analog computations are independent of a specific implementation in the same way as are digital computations. Further, analog computations can support interpretations in the same way as can digital computations (a point elaborated upon below).

1.3 In the theory of computation we study the properties of *idealized computational systems*. They are idealized because they make certain *idealizing assumptions*, which we expect to be only approximately instantiated in reality. For example, in the traditional theory of discrete computation, we make such assumptions as that tokens can be unambiguously separated from the background, and that they can be unambiguously classified as to type.

1.3.1 The theory of discrete computation has been well developed since the 1930s and forms the basis for contemporary symbolic approaches to cognitive modeling. In contrast, though exploration of continuous computation has been neglected until recently, we expect that continuous computational theory will provide a foundation for connectionist cognitive models (MacLennan 1988, in press-a, in press-b). Although there are many open questions in this theory — including the proper definition of computability, and of universal computing engines analogous to the Universal Turing Machine — the general outlines are clear (MacLennan 1987, 1990c, in press-a, in press-b; Wolpert & MacLennan submitted; see also Blum 1989; Blum & al. 1988; Franklin & Garzon 1990; Garzon & Franklin 1989, 1990; Lloyd 1990; Pour-El & Richards 1979, 1981, 1982; Stannett 1990).

1.3.2 In general, a *computational system* is characterized by: (1) a *formal* part, comprising a state space and processes of transformation; and (2) an *interpretation*, which (a) assigns meaning to the states (thus making them *representations*), (b) assigns meaning to the processes, and (c) is *systematic*. For continuous computational systems the state spaces and transformation processes are continuous, just as they are discrete for discrete computational systems. Systematicity requires that meaning assignments be continuous for continuous computational systems, and compositional for discrete computational systems (which is just continuity under the appropriate topology).

1.4 Whether discrete or continuous computation is a better model for cognition is a significant empirical question. Certainly connectionism shows great promise in this regard, but it leaves open the question of how representations get their meaning. The foregoing shows, I hope, that the continuous/discrete (or analog/digital) computation issue is not essential to the symbol grounding problem. I don't know if Harnad is clear on this; sometimes he seems to agree, sometimes not. What, then, is essential to the problem?

2.0 How Do Representations Come to Represent?

2.1 After contemplating the Chinese Room Argument for about a decade now, I've come to the conclusion that the "virtual minds" form of the Systems Reply is basically correct. That is, just as a computer may simultaneously be several different programming language interpreters at several different levels (e.g. a machine language program interpreting a LISP program interpreting a PROLOG program), and thereby instantiate several virtual machines at different levels, so also a physical system could simultaneously instantiate several minds at different levels. There is no reason to suppose that these "virtual minds" would have to be aware of one another or that the system would exhibit anything like multiple personality disorder. Nevertheless, Harnad offers no argument against the virtual minds reply, although perhaps we are supposed to interpret his summary dismissal ("unless one is prepared to believe," 4.2) as an argument *ad hominem*. He admits in Hayes & al. (1992) that it is a matter of intuition rather than of proof.

2.2 However, I agree with Harnad and Searle that symbols do not get their meanings merely through their formal relations with other symbols, which is in effect the claim of computationalism (analog or digital). In this sense, connectionist computationalism is no better than symbolic computationalism.

2.2.1 There is not space here to describe an alternate approach to these problems, but I will outline the ideas and refer to other sources for the details. Harnad argues that there is an "impenetrable 'other-minds' barrier" (Hayes & al. 1992), and from a philosophical standpoint that may be true, but from a scientific standpoint it is not. Psychologists

and ethologists routinely attribute "understanding" and other mental states to other organisms on the basis of external tests. The case of ethology is especially relevant, since it deal with a range of mental capabilities, which, it's generally accepted, includes understanding and consciousness at one extreme (the human), and their absence at the other (say, the amoeba). Therefore it becomes a scientific problem to determine whether an animal's response to a stimulus is an instance of it understanding the *meaning* of a symbol or merely responding to its physical form (Burghardt 1970; Slater 1980).

2.2.2 Burghardt (1970) solves the problem of attributing meaning to symbols by defining communication in terms of behavior that tends to influence receivers in a way that benefits the signaller or its group. Although it may be difficult in the natural environment to reduce such a definition to operational terms, the techniques of *synthetic ethology* allow carefully-controlled experimental investigation of meaningful symbol use (MacLennan 1990b, 1992; MacLennan & Burghardt submitted). (For example, we've demonstrated the evolution of meaningful symbol use from meaningless symbol manipulation in a population of simple machines.)

2.3 Despite our differences, I agree with Harnad's requirement that meaningful symbols be grounded. Furthermore, representational states (whether discrete or continuous) have sensorimotor grounding, that is, they are grounded through the system's interaction with its world. This makes transduction a central issue in symbol grounding, as Harnad has said.

2.3.1 Information must be materially instantiated — represented in a configuration of matter and energy — if it is to be processed by an animal or a machine. A *pure transduction* changes the kind of matter or energy in which information is instantiated. Conversely, a *pure computation* changes the configuration of matter and energy — thus processing the information — without changing its material embodiment. We may say that in transduction the *form* is preserved but *substance* is changed. In computation, in contrast, the *form* is changed but the *substance* remains the same. (Most actual transducers do not do pure transduction, since they change the form as well as the substance of the information.)

2.3.2 Observe that the issue of transduction has nothing to do with analog/digital (continuous/discrete) computation question; transduction can be either continuous or discrete depending on the kind of information represented. Continuous transducers transfer an image from one space of continuous physical variables to another; examples include the retina and robotic sensor and effector systems. Discrete transducers transfer a configuration from one discrete physical space to another; examples include photosensitive switches, toggle switches, and on/off pilot lights.

2.3.3 Harnad seems to be most interested in continuous-to-discrete transduction, if we interpret his 'analog world' to mean the world of physics, which is dominated by continuous variables, and we assume the output of the transducers are discrete symbols. The key point is that the specific material basis (e.g. light energy) for the information "out there" is converted to the unspecified material basis of formal computation inside the computer. Notice, however, that this is not pure transduction, since in addition to changing the substance of the information it also changes its form; in particular it must classify the continuous image in order to assign it to one of the discrete symbols, and so we have computation as well as transduction. (We can also have the case of an "impure" discrete-to-continuous transduction; an example would be an effector that interpolates between discretely specified states. Impure continuous/continuous and discrete/discrete transducers also occur; an analog filter is an example of the former.)

3.0 Conclusions

3.1 Harnad's notion of symbol grounding is an important contribution to the explanation of intentionality, meaning, understanding and intelligence. However, I think he confuses things by mixing it up with several other, independent issues. One is the important empirical question of whether discrete or continuous representational spaces and processes — or both or neither — are a better explanation of information representation and processing in the brain. The point is that grounding is just as important an issue for continuous (analog) computation as for discrete (digital) computation. Second, Harnad ties the necessity of symbol grounding to Searle's Chinese Room Argument with its problematic appeal to

consciousness. This is unnecessary, and in fact he makes little use of the Chinese Room except to argue for the necessity of transduction. There is no lack of evidence for the sensorimotor grounding of meaningful symbols. Given the perennial doubt engendered by Searle's argument, I would prefer to depend upon a more secure anchor.

REFERENCES

Blum, L. (1989). *Lectures on a theory of computation and complexity over the reals (or an arbitrary ring)* (Report No. TR-89-065). Berkeley, CA: International Computer Science Institute.

Blum, L., Shub, M., & Smale, S. (1988). On a theory of computation and complexity over the real numbers: NP completeness, recursive functions and universal machines. *The Bulletin of the American Mathematical Society*, **21**, 1–46.

Burghardt, G. M. (1970). Defining 'communication'. In: J. W. Johnston Jr., D. G. Moulton and A. Turk (Eds.), *Communication by Chemical Signals* (pp. 5–18). New York, NY: Century-Crofts.

Franklin, S., & Garzon, M. (1990). Neural computability. In O. M. Omidvar (Ed.), *Progress in neural networks* (Vol. 1, pp. 127–145). Norwood, NJ: Ablex.

Garzon, M., & Franklin, S. (1989). Neural computability II (extended abstract). In *Proceedings, IJCNN International Joint Conference on Neural Networks* (Vol. 1, pp. 631–637). New York, NY: Institute of Electrical and Electronic Engineers.

Garzon, M., & Franklin, S. (1990). Computation on graphs. In O. M. Omidvar (Ed.), *Progress in neural networks* (Vol. 2, Ch. 13). Norwood, NJ: Ablex.

Hayes, P., Harnad, S., Perlis, D. & Block, N. (1992). Virtual symposium on the virtual mind. *Minds and Machines* (in press)

Lloyd, S. (1990). *Any nonlinearity suffices for computation* (report CALT-68-1689). Pasadena, CA: California Institute of Technology.

MacLennan, B. J. (1987). Technology-independent design of neurocomputers: The universal field computer. In M. Caudill & C. Butler (Eds.), *Proceedings, IEEE First International Conference on Neural Networks* (Vol. 3, pp. 39–49). New York, NY: Institute of Electrical and Electronic Engineers.

MacLennan, B. J. (1988). Logic for the new AI. In: J. H. Fetzer (Ed.), *Aspects of Artificial Intelligence* (pp. 163–192). Dordrecht, NL: Kluwer Academic Publishers.

MacLennan, B. J. (1989). *The Calculus of functional differences and integrals* (Technical Report CS-89-80). Knoxville, TN: Computer Science Department, University of Tennessee.

MacLennan, B. J. (1990a). *Functional Programming Methodology: Practice and Theory.* Reading, MA: Addison-Wesley.

MacLennan, B. J. (1990b). *Evolution of communication in a population of simple machines* (Technical Report CS-90-99). Knoxville, TN: Computer Science Department, University of Tennessee.

MacLennan, B. J. (1990c). *Field computation: A theoretical framework for massively parallel analog computation; parts I – IV* (report CS-90-100). Knoxville, TN: University of Tennessee, Computer Science Department.

MacLennan, B. J. (1992). Synthetic ethology: An approach to the study of communication. In: C. G. Langton, C. Taylor, J. D. Farmer and S. Rasmussen (Eds.), *Artificial Life II* (pp. 631–658). Redwood City, CA: Addison-Wesley.

MacLennan, B. J. (in press-a). Continuous symbol systems: The logic of connectionism. In Daniel S. Levine and Manuel Aparicio IV (Eds.), *Neural Networks for Knowledge Representation and Inference.* Hillsdale, NJ: Lawrence Erlbaum.

MacLennan, B. J. (in press-b). Characteristics of connectionist knowledge representation. *Information Sciences*, to appear.

MacLennan, B. J., & Burghardt, G. M. (submitted). Synthetic ethology and the evolution of cooperative communication.

Pour-El, M. B., & Richards, I. (1979). A computable ordinary differential equation which possesses no computable solution. *Annals of Mathematical Logic*, **17**, 61–90.

Pour-El, M. B., & Richards, I. (1981). The wave equation with computable initial data such that its unique solution is not computable. *Advances in Mathematics*, **39**, 215–239.

Pour-El, M. B., & Richards, I. (1982). Noncomputability in models of physical phenomena. *International Journal of Theoretical Physics*, **21**, 553–555.

Slater, P. J. B. (1983). The study of communication. In: T. R. Halliday and P. J. B. Slater (Eds.), *Animal Behavior Volume 2: Communication* (pp. 9–42). New York, NY: W. H. Freeman.

Stannett, Mike (1990). X-machines and the halting problem: Building a super-Turing machine. *Formal Aspects of Computing*, **2**, 331–341.

Wolpert, D., & MacLennan, B. J. (submitted). A computationally universal field computer which is purely linear.