Computers and Computation in Cognitive Science

Tim van Gelder
Department of Philosophy
University of Melbourne
Parkville 3052 Australia
tgelder@ariel.unimelb.edu.au

Appearing in: M. T. Michalewicz ed., *Advances in Computational Life Sciences Vol.2: Humans to Proteins.* Melbourne: CSIRO Publishing.

Abstract
Digital computers play a special role in cognitive science—they may actually be instances of the phenomenon they are being used to model. This paper surveys some of the main issues involved in understanding the relationship between digital computers and cognition. It sketches the role of digital computers within orthodox computational cognitive science, in the light of a recently emerging alternative approach based around dynamical systems.

Key words: digital computers, computation, cognition, systems, representation, modeling, dynamical systems.

1. Introduction

Computers play many roles in cognitive science. Most are the ordinary sorts of roles they play in many other sciences as well——statistical analysis, visualisation, email, and so on. However, due to the special nature of the subject matter of cognitive science, computers play roles quite unique to that field. According to the dominant theory, cognition is a form of digital computation. Cognitive agents such as ourselves are widely held to be computers. Thus, a digital computer can be a model of cognition in a very literal sense—much like a scale model of a real bridge, or a laboratory mouse in a medical experiment. In other words, the digital computer can actually be an instance of the phenomena under study. In meteorology, a digital computer simulating a thunderstorm is not itself a thunderstorm. In cognitive science, by contrast, a digital computer modeling human reasoning processes may well be carrying out reasoning.

One of the major challenges posed by the use of computers in cognitive science has been that of achieving an adequate philosophical understanding of this uniquely intimate theoretical relationship between cognition and computers. In recent years, this issue has come to the forefront as the dominant theory has been challenged by the emergence of an alternative approach, within which cognitive systems are taken to be dynamical systems rather than computers. Even within this alternative approach, however, computers play crucial theoretical roles, and there are now fascinating new questions concerning the relation between dynamical systems and computers on one hand, and dynamical systems and cognition on the other.

This paper offers a brief overview of the some of the major issues in this area, as well as the directions of current debate. Since the main point of the paper is to discuss the roles of computers in two very different kinds of modeling, it begins with some stage-setting introducing very general notions such as cognition, systems, models, etc..

2. Modeling, Computers and Dynamical Systems in Cognitive Science

2.1 What is Cognitive Science?

Cognitive science is often described as the science of the mind, or the mind / brain. However, on any reasonably narrow construal of the term “cognitive science,” there will always be numerous aspects of the mind / brain with which cognitive science is not concerned. A better approach is to regard cognitive science as the interdisciplinary science of *cognition*, where cognition is the totality of mechanisms, states, processes, etc., causally responsible for a certain range of sophisticated behaviours exhibited most notably by people, and to a lesser extent by other animals and some artefacts—behaviors such as perception, reasoning, linguistic performance, problem solving, planning, and acting. One of the most distinctive features of cognition is that it typically involves coordination of behavior with respect to some remote domain, even in the absence of direct causal connection with that domain. For example, devising a plan for carrying out a series of shopping errands is a matter of getting one’s thinking processes here and now to conduct themselves appropriately with respect to states of affairs which are remote in a number of ways—far away in space, in the future, and which perhaps will never even take place.

2.2 Representation

How is it that systems manage to exhibit behaviours of these rather special kinds? The standard answer is that they possess *knowledge* of the relevant domains stored in the form of internal representations which guide the internal processes so as to generate appropriate behaviour. Unfortunately, *representation* is one of the most vague and ambiguous notions in cognitive science, notwithstanding its foundational status. No specific characterisation is universally accepted in the field. The version recommended here is adapted from an informal definition offered by in (Haugeland, 1991). The intuitive idea is that a representation is something that “stands in” for some relevant feature of the domain. Expanding this idea a little, a representation is

1. some relatively discrete, localised and identifiable part or aspect of the system, which
2. corresponds to some (possible) feature of a domain, in virtue of
3. a scheme which systematically determines the correspondences between representations and features of that domain, and which
4. plays a causal role in allowing the system to coordinate its behavior with respect to that domain.

2.3 Systems

The term “system” is often used vaguely to refer to any complex thing we wish to talk about. In the remainder of this paper it will be used in a more restricted and precise way. A system is a set of entities—*variables*—all of which change interdependently over time. A *parameter* is an entity upon which change in the system depends, but whose own change does not depend on the system. The *state* of a variable at a time is its value at that time; the state of a system is the set of states of its variables.

Variables, and so systems themselves, can be concrete or abstract. Concrete variables are aspects of the actual world. Abstract entities are not concrete, but neither are they merely fictional or only "in the mind". A satisfactory account of the nature of abstract entities is a difficult philosophical problem, but for the moment it will be convenient to think of them as inhabitants of Plato's Heaven, the abstract domain of pure form or structure, which includes such things as the number 2, the minimax search algorithm and the Moonlight sonata.

Change is always change over time, and so every system must inhabit time in some sense or other. For current purposes, time is best thought of as an ordered set of entities. An ordered set is a time set insofar as it serves to provide orderings over another set. *Real* time is the time in which all concrete variables change; it is the set of instants of time, ordered by temporal priority. Abstract variables do not change in real time, for Plato's Heaven is "eternal" in the sense of standing outside real time. The time set for abstract systems is always another set of abstract entities, such as the integers or the real numbers.

2.4 Classical Dynamical Modeling

The most famous system in the history of science is undoubtedly "the" solar system. In fact, there is no single solar system. There is to be sure just one collection of planets, etc., revolving around our sun. However in the vicinity of that collection there are very many distinct sets of interdependently changing variables. Out of these we must select the system we choose to study. A classic choice is the set of positions and velocities of the sun and planets. These change over time in a way that conforms closely to the laws of classical mechanics as first formulated by Newton and Laplace. To understand this relationship between the concrete solar system and the abstract mathematical laws we must bring abstract systems and modeling into the picture.

Suppose we have two abstract vector variables. We could refer to them with arbitrarily chosen symbols such as *x* and *y*, but it will be more illuminating to use related vector variables, $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$. Suppose additionally that the values of these variables happen to change interdependently in a way that is governed by the differential equation

$$\ddot{\boldsymbol{r}} = -\frac{G(m_1 + m_2)}{r^2}\hat{\boldsymbol{r}}$$

where G, m_1 and m_2 are arbitrarily chosen parameters. Then $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ constitute an abstract system. A pure mathematician might be interested to explore the properties of this system considered in its own right.

An explanatory *target* is some complex entity, phenomenon, etc., we wish to understand better. Given a target, a *model* is some other thing which is

1. similar in relevant respects;
2. well understood already, or for whatever reason more amenable to understanding than the target; and
3. used by us, in virtue of 1. and 2., to assist us in understanding the target.

In natural science, the target is some natural phenomenon such as the motions of the planets, and we often use abstract systems as models. The motion of the earth around the sun can be understood by relating it to the abstract mathematical system just specified. We relate the concrete and abstract systems by setting up correspondences between their variables:

$\boldsymbol{r}$ <=> the position of the earth relative to the sun

$\dot{\boldsymbol{r}}$ <=> the velocity of the earth relative to the sun

G <=> a constant of gravitational attraction

m_1 <=> the mass of the sun

m_2 <=> the mass of the earth

Note that, strictly speaking, the symbols $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ do not refer to the these concrete variables directly, and the equation does not govern the behaviour of relative position and velocity directly[1].

Models are almost never perfect. A model must be understandable, and this often demands that it be much simpler than the target. The criterion of goodness of a model is not faultless and exhaustive similarity to the target, but rather that it advance our understanding of the target.

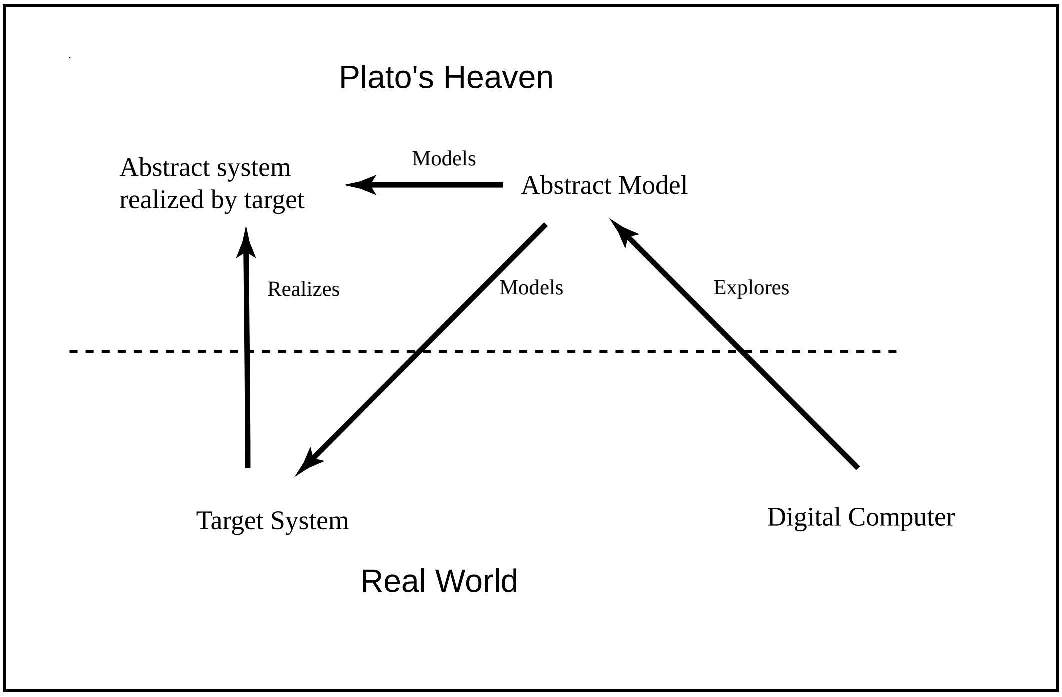
Model variables must evolve over time in a way that is a reasonably close match to the evolution of the target variables with which they correspond. For evaluating this match, it is most convenient if the equations which govern the model take the form of functions of time. In the case of a system governed by differential equations, these functions would constitute solutions to those equations. Unfortunately, for nonlinear systems it is often difficult or impossible to express the governing equations as functions of time. For example, Newton’s laws of planetary motion are nonlinear, and there are no general solutions in the case of three or more bodies. Nevertheless, using a combination of mathematical insight and much effort, approximations to the time-series specified by the solutions can be calculated. These calculations standardly involve a great deal of shuffling of mathematical symbols in accordance with algorithms.[2]

Any system which carries out these symbol shufflings is a digital computer, in the sense defined below, whether implemented by means of a human using pen and paper or in programmable electronic form. Consequently, there is a very important role for digital computers in classical dynamical modeling. They are used as tools for exploration of abstract dynamical models, and thereby the explanatory targets themselves.

---

1 For extensive discussion of classical dynamical modeling of the solar system, see (Griffiths, 1985). The simple differential equation governing the motion of two bodies given above appears on p. 147.

2 They might, in fact, involve analog rather than digital computation. This distinction is discussed below.



2.5 Dynamical Systems.

The Newtonian model of planetary motion is a *dynamical* model. Both the abstract model and the concrete system being modelled are *dynamical* systems. What does this mean?

The term "dynamical system" has come to be used many different ways across the various disciplines in which it occurs. At one extreme, dynamical systems are taken to be sets of bodies moving under the influence of forces. At the other extreme, they are taken to be anything that changes, i.e., any system in the current sense. Table 1 illustrates this impressive diversity.

Guiding Idea	Examples
1. A system of bodies whose motions are governed by forces. Such systems form the domain of dynamics considered as a branch of classical mechanics.	"a collection of a large number of point particles." (Desloge, 1982) p.215 Webster's: "dynamics...a branch of mechanics that deals with forces and their relation primarily to the motion... of bodies of matter."
2. A physical system whose state variables include rates of change	"In the original meaning of the term a dynamical system is a mechanical system with a finite number of degrees of freedom. The state of such a system is usually characterized by its position...and the rate of change of this position, while a law of motion describes the rate of change of the state of the system." (1989) p.328
3. A system of first-order differential equations; equivalently, a vector field on a manifold	a dynamical system is "simply a smooth manifold *M*, together with a vector field *v* defined on *M*." (Casti, 1992) p.109
4. Mapping on a metric space	"A *dynamical system* is a transformation f:**Z**→**Z** on a metric space (**Z**, *d*)." (Barnsley, 1988) p.134.
5. State-determination	"a dynamical system...is one whose state at any instant determines the state a short time into the future without any ambiguity." (Cohen & Stewart, 1994) p.188
6. Any mapping, equation, or rule.	A dynamical system may be defined as a deterministic mathematical prescription for evolving the state of a system forward in time." (Ott, 1993) p.6
7. Change in time	"A dynamical system is one which changes in time." (Hirsch, 1984) p.3 "The term *dynamic* refers to phenomena that produce time-changing patterns...the term is nearly synonymous with time-evolution or pattern of change." (Luenberger, 1979) p.1

Table 1. Various definitions of "dynamical system".

The significance of this semantic chaos for the current discussion is that there is no single definitively correct definition of the term "dynamical system." In that sense, it is impossible to say what dynamical systems "really are." Rather, the challenge is to select or fashion a characterisation which is maximally useful in helping us to understand the theoretical issues central to current disputes in cognitive science.

Two superficially obvious features of the use of dynamical systems in cognitive science can serve as guides. The first is that dynamical systems are standardly regarded as being quite different in nature to ordinary digital computers. The second is that in dynamical models in cognitive science the variables are always numerical. One important reason for this is that numerical variables have quantitative properties (i.e., it makes sense to talk of the distance between any two values of a variable, or between any two states of the system). These considerations suggest defining dynamical systems as *quantitative* systems; roughly, systems in which there are distances, and these distances matter to behavior. This can be true in progressively deeper ways, giving rise to progressively more substantial senses in which a system can count as dynamical.

- *Quantitative in state.* First, there can be distances between any two overall states of the system, such that the behavior of the system depends on these distances. More precisely, a system is quantitative in state when there is a metric[3] over the state set, such that behavior is systematically related to distances as measured by that metric. Such systems will be governed by a rule compactly specifying this distance-dependent change.

  Standardly, state sets are metric spaces because the variables of the system are quantities, i.e., are themselves such that there is a metric over their sets of values. Quantities can be either abstract or concrete. For example, the variables $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ of the model of the previous section are abstract quantities; they correspond via measurement to the relative position and velocity of earth and sun, which are concrete quantities.
- *Quantitative state/time interdependence.* A system is quantitative in *time* when time is a quantity, i.e., there is a metric over the time set, such that system behavior is systematically related to distances as measured by that metric. At least in cognitive science practice, systems that are quantitative in time are also quantitative in space, and these properties are interdependent. That is, the behavior of the system is such that *amounts* of change in state are systematically related to *amounts* of elapsed time. Such systems are governed by a rule specifying a quantitative relationship between change in state, elapsed time, and current state. In concrete systems, this rule captures causal organisation; that is, the system changes the way it does because system variables have the quantitative properties in terms of which the rule is expressed. When both state and time are quantitative, the system exhibits *rates* of change. Systems that are interdependently quantitative in state and time are governed by rules specifying the rate of change in terms of current state (i.e., differential equations).
- *Rate dependence.* Third, some systems are such that their rates of change depend on current rates of change. In these systems, variables include both basic variables and the rates of change of those variables. The solar system is a classic example. Systems whose behavior is governed by rules most compactly expressed as sets of higher-order differential equations are quantitative in this sense.

In what follows, a system is taken to be dynamical to the extent that it is quantitative in one of the above senses.[4]

2.6 Digital computers

Fortunately, there is a much greater level of agreement in cognitive science and elsewhere over the notion of digital computer.

A *computer* is simply anything that computes in some way or other. *Computing* is an informal notion; the basic idea is that of a process systematically transforming “questions” into “answers”— inputs into outputs, start states into

---

[3] A metric over a set X is a function d:$X\times X \rightarrow \mathbf{R}$ that assigns to every pair of elements x and y a number $d(x,y)\geq 0$ such that $d(x,y)=0$ iff $x = y$, $d(x,y) = d(y,x)$, and $d(x,y)\leq d(x,z) + d(z,y)$.

[4] This formulation is designed to accommodate some rather special cases of dynamical systems whose behavior is generally quantitative except at certain isolated points (Gregson, 1993; Zak, 1990).

final states, etc.. The *function* computed by that process is the set of question / answer pairs themselves (*formal* computation), or the set of pairs of entities they represent (*semantic* computation). Now, in this general sense pretty much anything can be construed as a computer. Computation only gets interesting when significant constraints are placed on the kinds of processes involved. In classical computation theory, the standard approach has been to require that processes be *effective*, i.e., produce their results by means of a finite number of basic operations specified by an algorithm (a finite recipe, or set of instructions specifying basic operations).

Digital computers, in the sense that matters for cognitive science, are those systems which carry out effective semantic computation. That is, they are systems whose behaviors constitute algorithmically specified finite sequences of basic operations over representations. This characterisation can be broken down into four fundamental requirements on a system to count as a digital computer:

- *Digital variables and states*. First, for each variable there must be some set of *discrete* values which the variable instantiates *digitally* for the purposes of system behavior. In the concrete case, this means that the variable must instantiate those variables positively and reliably.[5] When all variables in a system are digital, the system's *states* are also digital. The basic operations required by effective computation correspond to digital state transitions.
- *Time as discrete order*. The time set must be a discrete order whose elements are the times at which the system digitally occupies its states. In abstract systems, this is usually the positive integers. In concrete systems, it is the set of periods of real time at which the machine digitally instantiates its states, as rendered discrete by the flux of transition between states. These are indexed by the positive integers (t_1, t_2, etc.).
- *Formal rule*. Effective computation requires basic operations to be specified by an algorithm. For a system to carry out effective computation, its behaviors must be governed by a formal rule, i.e., a finite rule specifying state transitions solely on the basis of digital properties of states. In concrete systems, this rule must capture one level of causal organisation. That is, the transitions described by the rule must happen the way they do *because* the states bear the digital properties in terms of which the rule is expressed.
- *Interpretation*. The system's states and behaviors must yield to systematic interpretation. That is, there must be some domain, and correspondences between the system and the domain, such that (a) the correspondences are *systematic* with respect to those digital aspects of the system in terms of which the rule governs system behavior, and (b) the system's states and behaviors *make sense* in the light of those correspondences.[6]

---

5 See (Haugeland, 1985), Chapter 2. In abstract systems, discreteness of values suffices for digitality.

6 What is it to "make sense"? This is a difficult issue; see (Haugeland, 1985), Chapter 3, for discussion. Every digital system can be set up in systematic correspondence with some domain (such as integers and functions over them) but not all such systems have an interpretation in the current sense. The ones that do are those exhibiting a further kind of order that does or could seem

Any (interpreted) Turing Machine constitutes a prototypical example of a digital computer. A Turing Machine has two major components. The first is a tape of squares or cells in which symbols can be written. The second is a "head"—a mechanism which can move up and down the tape changing the symbols in the squares and its own state. Considered as a formal *system*, a Turing Machine consists of a large set of formal variables which include the tape squares, the head state, and the current square (head location). Change in the system is a matter of discrete transitions from one overall configuration of tape and head to another. These transitions are governed by a set of rules which specify changes to the overall state by specifying changes in the state variables corresponding to head location, the head state and the head location in terms of current values of those variables. These rules can be succinctly summarised in the form of a table known as the machine table.

One standard way to understand Turing Machines as computing is to consider a machine whose tape squares are either empty or contain the symbol 1. A string of non-empty squares of length n is interpreted as representing the number n. A suitably constructed machine will start with such a string and halt having produced another string of length m. It has then computed the mapping from n to m, which is one argument-value pair of a function defined over the integers.

2.7 The Computational Hypothesis & Computational Modeling.

The guiding idea of mainstream cognitive science is that cognitive systems are digital computers in the general sense just described. (They need not, of course, be Turing Machines specifically, or very similar to ordinary electronic digital computers.) The classic formulation of this idea is due to Allen Newell and Herbert Simon:

> *The Physical Symbol System Hypothesis.* A physical symbol system has the necessary and sufficient means for general intelligent action. (Newell & Simon, 1976)

"Physical symbol systems" is just an alternative name for digital computers in the sense outlined above. If we generalise the notion of general intelligent action to include all cognitive performance, and view this hypothesis in the context of the attempt to scientifically *understand* cognitive systems, Newell and Simon's hypothesis becomes:

> *The Computational Hypothesis:* Sophisticated cognitive systems are digital computers, and can be scientifically understood as such.

This hypothesis implies immediately that digital computers will occupy a very special role in modeling in cognitive science. If cognition is just digital computation, then digital computers can exhibit cognition. Therefore, a digital computer in cognitive science can (in principle at least) actually instantiate the phenomena under study. This means that the digital computer can go beyond simply being a tool for simulation, etc.. It can be "the real McCoy." Therefore, a concrete digital computer can be used to model cognition in a way that it can't be used to model, say, a

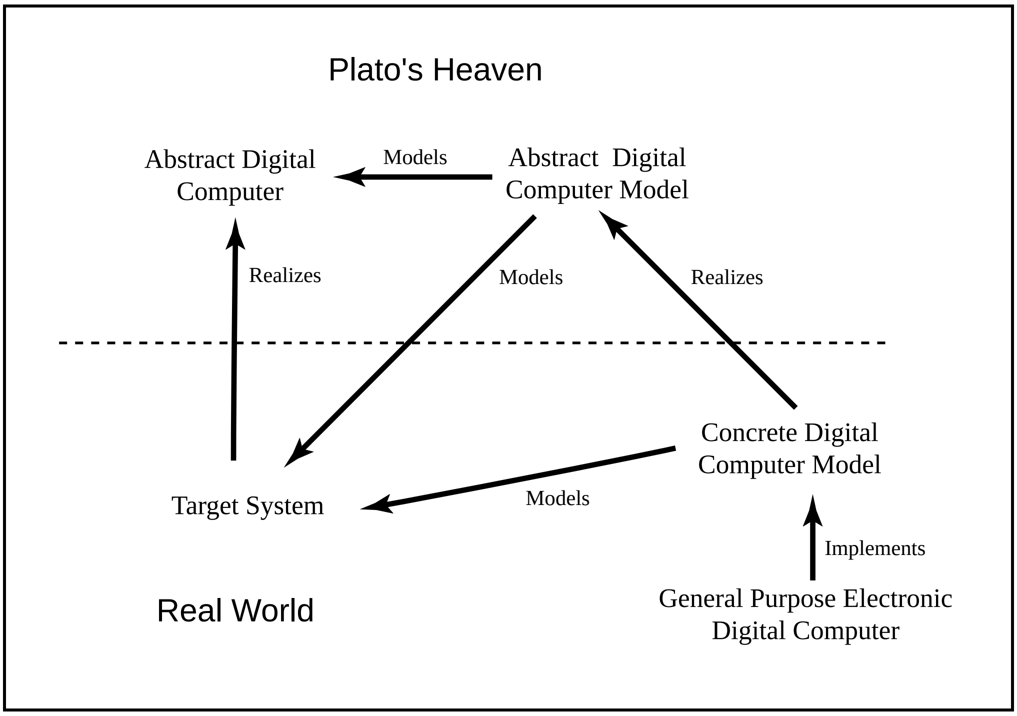
---

patterned or reasonable *to us* (humans); thus, whether something is a digital computer is human-relative.

Note that having an interpretation in the current sense may not be enough to guarantee that the system has "meaning" in some stronger sense, (and hence, perhaps, "mind"). For discussion of these issues, see (Harnad, 1990; Searle, 1980).

thunderstorm. In cognitive science we set up the digital computer as an *example* of the process we are trying to study.

The basic theoretical structure of computational modeling of cognition is outlined in Figure 2.



2.8 Virtues of the Computational Hypothesis

The Computational Hypothesis is an empirical hypothesis, in the sense that it might be false, and only sustained scientific investigation can determine whether or to what extent it is true. A vast amount of empirical work has already been conducted within the general framework defined by the Computational Hypothesis. Some of this work has been impressively successful, but much of it has encountered grave difficulties. The jury is definitely still out on the extent to which the Computational Hypothesis is true, and probably will remain out for a long time yet. Nevertheless, despite this empirical uncertainty, there are a number of profound general reasons why it is especially attractive.

- *Meaning*. Recall that cognitive systems are distinctive in that they manage to coordinate their behaviour with respect to further domains, including remote and non-actual domains, and the standard strategy for explaining how they are able to do this is to suppose that they contain internal representations of those further domains. Digital computers provide a relatively clear and well-worked-out understanding of how natural (physical, mechanistic) systems can contain representations and have their behaviour guided by representations.
- *Universality*. Digital computers can carry out any process which reduces to a finite number of finite operations. More precisely, on the standard definition of computation (see below for further discussion) the functions that are computable by digital computers are those which are, or are equivalent to, partial recursive functions over the integers. Thus not every function is computable, but a very large

number of functions are. An even more remarkable fact is that many digital computers are provably *universal*. That is, they can compute any function that is computable by any other computer whatsoever, provided only that they begin with a suitable configuration of symbols (i.e., are programmed to imitate that other machine).

The fact that certain computers are universal suggests that any particular kind of cognitive performance can be carried out by such a computer, just as long as any instance can be reduced to a finite number of finite operations. In the absence of any competing hypothesis, this provides good reason to believe that cognitive systems must be digital computers.

An additional general empirical fact strengthens this line of thought. As René Descartes noted, people exhibit a very wide range of capacities. They can do things as diverse as walk around the room, conduct conversations, plan their activities many years into the future, and do mathematics. Human cognition is a particularly difficult scientific challenge because it appears to subserve such a variety of activities. The universality of digital computers provides a powerful way to understand how such variety is possible.

- *Complexity*. Sophisticated cognitive performances are presumed to demand highly complex mechanisms. Because of their digital nature, digital computers sustain extremely high levels of complexity. (The universal digital computer on which this paper is being written—a standard commercial PC—is a case in point.) In the absence of any competing account of how the complexity of cognitive systems can be sustained, it is plausible that sophisticated cognitive systems must be digital computers.
- *Systematicity*. A cognitive capacity is systematic insofar as exhibiting some forms of that capacity implies exhibiting other forms. Many aspects of cognition exhibit a high degree of systematicity. For example, in normal cases the ability to multiply 47 and 63 implies the ability to multiply 47 and 65; anything that can do one can do the other. The assumption that cognitive systems are digital computers provides one way to explain such systematicity (See (Fodor & Pylyshyn, 1988) for elaboration of this point.)
- *What else could it be?* Until at least the mid-1980s, there were no general hypotheses as to the nature of cognitive systems that could stand as serious competitors to the Computational Hypothesis in scope, power, and precision.

2.9 The Dynamical Hypothesis

Over the last decade or more the dominance of mainstream computational cognitive science has been challenged by the rapid growth of a research paradigm within which cognitive systems are taken to be in the first instance *dynamical* systems. The theoretical core of this paradigm is

> *The Dynamical Hypothesis:* Natural cognitive systems are dynamical systems, and can be understood as such.[7]

Note that the Dynamical Hypothesis is concerned primarily with *natural* cognition—that is, cognition in humans and other animals. It is an empirical hypothesis about the nature of natural cognition. It can happily allow artificial intelligence might

[7] For elaboration of this hypothesis, see (van Gelder, 1995; van Gelder, forthcoming).

construct digital computers which exhibit cognitive capacities even though they operate in a fundamentally different way. This in fact seems to be the case with chess-playing computers—they can now play first-rate chess, but the internal processes by which they come up with good moves are very different from those used by human masters.

The structure of modeling practice in dynamical cognitive science is basically the same as in classical dynamical modeling (Figure 1). Note that digital computers are part of this structure. They play much the same role in dynamical cognitive science as they do in classical dynamical modeling: that is, they are tools in the development and exploration of abstract dynamical system models. In practice, the digital computer has been absolutely essential to the development of dynamical cognitive science. Without digital computers for use in simulation, graphical displays, etc., it would have been impossible to come to an adequate level of understanding of the properties of nonlinear dynamical systems, and so it would have been impossible to use such systems as models of complex cognitive processes. This is why dynamical cognitive science has only flourished *after* substantial amounts of computing power became available for use by cognitive scientists. The origins of the dynamical hypothesis can be traced all the way back to David Hume in the eighteenth century, but it is only in the last decade or two that computing power, software packages and developments in dynamical systems theory made dynamical cognitive science feasible.

There is one important difference between dynamical modeling in cognitive science and classical dynamical modeling. The former faces an additional burden of explaining the cognitive system "as cognitive." Putting the point somewhat simplistically, classical dynamical modeling aims to describe and explain the changes that take place in some set of concrete variables. Dynamical cognitive science must do this, but also integrate any such account with a story about how those changes constitute cognitive performances, in the sense outlined above (s.2.1). Consider, for example, the difference between a dynamical model of brain temperature and blood flow and a dynamical model of decision making. Both must describe the change over time in some set of variables, but in the latter case, those changes must count as making a decision, i.e., must be related to some further domain of options, preferences, choices, and so forth.[8]

2.10 Connectionism

Connectionism is a form of cognitive science within which cognitive systems are modelled by means of neural networks. In this context, a neural network is a system with the following typical properties:

1. Variables are quantitative;
2. Variables are "neural" in the sense that, roughly, they change their "activation" in a way that depends on the activation of their neighbours and the strengths of the connections between them. In more general terms, change in a system variable is a function of some function of the values of a subset of system variables and a set of parameters known as connection weights. The first function is usually a sigmoid and the second is usually a weighted sum.

---

[8] See (Busemeyer & Townsend, 1993; Grossberg & Gutowski, 1987; Leven & Levine, 1996) for examples of dynamical models of decision making.

3. Variables are homogeneous, i.e., all variables take the same form. One equation schema suffices to describe the nature of change in all system variables, with only parametric differences between them.
4. The system is high dimensional, i.e., contains a relatively large number of variables.

Neural networks can instantiate digital computers, dynamical systems, or neither, depending upon whether they satisfy the further conditions for systems of those kinds. In fact, these days connectionist networks are overwhelmingly dynamical systems, and connectionist research based on such models is a special case of dynamical modeling of cognition. However, as we will see below, there are also generalised notions of computing according to which neural networks can be *both* dynamical systems and (non-classically) computational.

2.11 Virtues of the Dynamical Hypothesis

As with the Computational Hypothesis, there has already been considerable empirical work underneath the umbrella of the Dynamical Hypothesis (see (Port & van Gelder, 1995) for many different examples), but the extent to which the hypothesis is true is still very much an open issue. Many researchers have been attracted to dynamical cognitive science as response to difficulties encountered in developing adequate models within the mainstream computational paradigm. The dynamical approach does however possess a number of advantages in its own right:

- *Time.* All natural cognition unfolds in real time. The processes which make it up have a particular *timing*. There are speeds and rates of change. The success of many cognitive operations depends critically on getting the timing right in subtle ways. Dynamical models with continuous quantitative time offer the ability to model how cognitive processes unfold in time in much greater detail than digital computer models in which time is merely a discrete order.
- *Parallelism*. Cognitive systems often exhibit multiple simultaneous forms of change and interaction. For a crude example, consider playing a stroke in tennis. Perceptual and motor systems are running simultaneously and each is simultaneously affecting the other. Dynamical modeling provides a natural framework within which to accommodate parallel interactive processes.
- *Continuity.* Numerous cognitive processes appear to be highly sensitive to very subtle differences in the domain of concern. For example, a very slight variation in intonation can mean the difference between a sincere and a sarcastic utterance. Recognising this intonational shift matters greatly when it comes to making an appropriate response. These subtle differences must be reflected somehow in differences in the cognitive system. In digital computers, they must be symbolically represented. Dynamical systems with continuous variables can reflect these differences in subtle quantitative shifts of state. If the system is nonlinear, such shifts can then be responsible for appropriate differences in response.
- *Embeddedness.* Natural cognitive systems are embedded in the natural world. They are processes that take place when a brain is situated in a body which is itself situated in an environment. Any fully adequate account of cognition must explain how cognitive processes are related to the brain, the body and the environment. Dynamical forms of description are suitable for a great many aspects of the world in which natural cognition is embedded. For example, the behaviour of real neurons and their interaction are modelled by means of differential equations. An

advantage of the dynamical approach to cognition is that, by using the same conceptual tools to describe both cognition and that in which it is embedded, it minimises problems in explaining embedding.

- *Emergence*. Somehow, it has come about that *there are* natural cognitive systems. Evolution can provide only part of the answer. Just as there is a biological problem of physiological morphogenesis—how does the fantastic complexity of the human body develop from a single cell?—there is a problem of cognitive morphogenesis. How does it happen that raw biological (e.g., neural) materials come to be organised in the form of a complex cognitive system? This is a special case of a general scientific problem of the emergence of structure and organisation. Within dynamical systems theory, progress is being made in understanding how systems can "self-organise"—i.e., how large numbers of simple local interactions can result in the emergence of global order. Consequently, the dynamical approach to cognition appears to be a position to capitalise on this progress in explaining the emergence of cognition.

3. Some Current Debates

Thus far, this overview has situated digital computers in cognitive science in terms of a simple opposition between mainstream computational modeling and dynamical modeling. We now turn to some of the foundational issues that have arisen in the context of this opposition, much as the lightning is supposedly caused by friction between storm clouds.

3.1 Alternative Conceptions of Computation

Can dynamical systems compute? The standard notion of computation, generally known as *effective* computation, is the process of achieving some result in some finite number of finite operations specifiable by some finite schedule of rules. The Turing Machine framework was introduced as a means of formalising this intuitive notion. It turns out that every other formalisation of effective computation is equivalent to the Turing framework. For this reason, it is now generally accepted that effective computation reduces to whatever can be achieved by Turing Machines, or more generally by digital computers. Consequently, the answer to our question would appear to be distressingly simple: a dynamical system can compute if and only if it happens to instantiate a digital computer. It is a further question whether the conditions for counting as a dynamical system are compatible or incompatible with those for counting as a digital computer.

A more interesting issue is whether dynamical systems can be said to compute, not simply by instantiating digital computers, but in their own right. This will be possible only if there is some more general category of computation, such that standard effective or Turing computation is one species, and dynamical systems can instantiate some other species. To address this question, then, we need to clarify alternative notions of computation.

Over the past few decades, mathematicians and computer scientists have explored a great many alternatives to the classic Turing conception of computation. They can be coarsely sorted into the following two classifications:

- *Extended Effective Computation.* As mentioned above, the Turing-computable functions—and hence the effectively computable functions—are standardly taken to be all and only the partial-recursive functions over the integers. Many functions, including much of the subject matter of analysis and physics, do not belong to this category and hence are not computable in this basic sense. Much work (e.g., (Grzegorczyk, 1957)) has gone into extending the scope of effective computation to cover functions over the reals, complex numbers, etc.. The standard strategy is that of successive approximation. For example, a real number is Turing-computable on this approach if a Turing machine can compute in finite time a rational number which is within some arbitrarily chosen small distance of the real number. Note that there are functions which are not effectively computable even in this extended sense (Pour-El & Richards, 1989).

- *Non-effective computation*. In the most general sense, computation is a process which transforms in finite time some argument of a function (or representation thereof) into the corresponding value (or representation thereof). Computation is effective if the process reduces to a finite number of finite operations. There are various ways in which the effectiveness can be relaxed, giving rise to various kinds of non-effective computation. At the moment this is a rapidly developing field. Two currently prominent frameworks for mathematical formalisation of non-effective computation are:
  - *Analog Neural Networks*. One way to relax effectiveness is to maintain digital inputs and outputs, but to allow the process which transforms inputs to outputs to involve discrete-time change in real variables. One framework for the study of such computation is that of analog neural networks, as studied by Siegelmann & Sontag (Siegelmann & Sontag, 1994).
  - *Real Computation* Alternatively, one can also relax the requirement of digital inputs and outputs. Blum, Shub & Smale have studied “real computation”, in which (in their abstract form) take real numbers as inputs to a machine which is essentially an iterated mapping with a decision mechanism to halt the computation when some criterion is met (Blum, Shub, & Smale, 1989).

On the basis of these clarifications it is clear that dynamical systems can be understood as computational systems—i.e., as performing computation—even when not instantiating digital computers or carrying out effective computation. Indeed, it is now clear that various kinds of non-effective computational systems have super-Turing capacities, i.e., can compute functions which are not Turing-computable; this includes not only abstract mathematical systems, but physically realizable systems as well (Bournez & Cosnard, 1996).

3.2 Cognition and Non-classical Computation

The development of the theory of non-classical computation has opened a via media between purely dynamical models of cognition on one hand and mainstream computational accounts on the other. Previously it was common enough to talk of dynamical systems in vague or metaphorical terms as “information processing” or even “computing” (as in the term “computational neuroscience”) but such talk can now be put on a firm theoretical footing.

3.3 Emergent Computation

How is it that computation, whether digital or otherwise, comes about? How do we explain how the world came to be organised in just such a way, that some aspects of it happen to be performing computations? In the case of any artificial computer, there is a simple answer: they were built that way by us. (This is of course only a partial answer, for we can then ask how it came about that there are people able to build such things.) Obviously, however, the genesis of any “natural” computational system cannot be explained this way. Assuming that a Divine Creator is not an acceptable scientific hypothesis, the emergence of computation must be explained as a special case of the general problem of the emergence of complexity in nature. Currently, the dominant framework for such explanation is the theory of self-organisation under the broad umbrella of dynamical systems theory. This has given rise to the field of now known as “emergent computation”:

> Researchers in several fields have begun to explore computational models in which the behaviour of the entire system is in some sense more than the sum of the parts...In these systems interesting global behaviour *emerges* from many local interactions. When the emergent behaviour is also a computation, we refer to the system as an *emergent computation*.... (Forrest, 1991) p.1.

There is a natural fit between cognitive science and the field of emergent computation. On one hand cognitive systems may well be computational systems of some sort. On the other, the brain is a system comprised of a great many relatively simple components interacting on a local scale to form globally coherent behaviours. Nevertheless, it remains an open question whether the particular *kinds* of computation that constitute human cognition can be understood as emergent computation, i.e., whether cognition can be understood in emergentist terms. To date, the vast majority of sophisticated cognitive processes seem to lie well beyond the reach of emergentist techniques, suggestive and exciting as that work is.

3.4 Cognition and Representation

As mentioned in s.2.2 above, representationalism is the standard strategy for explaining how systems can exhibit interesting kinds of cognitive performance. Internal representations are the mechanisms which enable a cognitive system to coordinate its behaviour successfully with respect to other domains. However, it has become increasingly obvious that representations have associated costs: they must be produced, organised, maintained, and processed. Researchers in a number of fields have sought to understand cognition without paying these costs, i.e., how systems might exhibit some kinds of cognitive performance without containing any internal representations at all (e.g., (Brooks, 1991)).

The anti-representationalist trend in cognitive science dovetails nicely with dynamical cognitive science. Dynamics provides a very powerful set of tools for understanding complex systems without postulating any representations. Some kinds of dynamical systems can be understood as exhibiting a measure of cognitive performance, and yet when properly understood from a dynamical perspective can be seen to contain no representations at all (van Gelder, 1995). Researchers in cognitive science have produced non-representational dynamical models of various aspects of cognition (see, e.g., (Beer, 1995a; Beer, 1995b; Freeman & Skarda, 1990; Skarda & Freeman, 1987)). It is difficult to avoid considering the natural extrapolation of these developments—that the dynamical approach might make it possible to understand *all* cognition in non-representational terms. If this were true, it would be a very dramatic example of science overthrowing our ordinary ways of understanding ourselves and the world. A healthy debate has ensued, with participants sorting themselves into roughly four camps: the hard left, who are optimistic that representation-free cognitive science is just around the corner; moderate leftists, who argue that the anti-representational case is still largely open and worth exploring seriously ; the moderate right, who argue that dynamical approaches may force us to substantially reconceive representation, but not to reject it (e.g., (Clark, 1997)); and the conservative right, who stick steadfastly to the traditional represenationalist picture, cashed out in standard computational terms (e.g., (Fodor & Pylyshyn, 1988)).

4. Conclusion

Computation is essential to cognitive science, not merely as a practical aid in investigation, but as part of the theoretical fabric. However, the roles played by digital computers in cognitive science depend very much on the specific kind of cognitive science. In computational cognitive science, digital computers are used to instantiate processes of the very same kind as are found in natural cognition (if the theory is correct). In dynamical cognitive science, the role of digital computers is rather more similar to their role in any other science involving dynamical modeling of some natural phenomenon. Computation, cognition and dynamics form a rather complex theoretical tangle, and cognitive science is proceeding in cooperation with computer science and mathematics to sort out the various threads.

References

(1989) *Encyclopaedia of Mathematics*. Dordrecht: Kluwer.

Barnsley, M. (1988) *Fractals Everywhere*. San Diego: Academic Press, Inc.

Beer, R. D. (1995a) Computational and Dynamical Languages for Autonomous Agents. In R. Port & T. van Gelder ed., *Mind as Motion: Explorations in the Dynamics of Cognition*. Cambridge MA: MIT Press.

Beer, R. D. (1995b) A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, **72**, 173-215.

Blum, L., Shub, M., & Smale, S. (1989) On a Theory of Computation and Complexity Over the Real Numbers: NP Completeness, Recursive Functions and Universal Machines. *Bulletin of the American Mathematical Society*, **21**, 1-49.

Bournez, O., & Cosnard, M. (1996) On the computational power and super-Turing capabilities of dynamical systems. NeuroCOLT Technical Report Series No. NC-TR-96-005,

Brooks, R. (1991) Intelligence without representation. *Artificial Intelligence*, **47**, 139-159.

Busemeyer, J. R., & Townsend, J. T. (1993) Decision field theory: a dynamic-cognitive approach to decision making in an uncertain environment. *Psychological Review*, **100**, 432-459.

Casti, J. L. (1992) *Reality Rules: Picturing the World in Mathematics (Vols I, II)*. New York: J. Wiley.

Clark, A. (1997) *Being There: Putting Brain, Body and World Together Again*. Cambridge MA: MIT Press.

Cohen, J., & Stewart, I. (1994) *The Collapse of Chaos*. New York: Penguin.

Desloge, E. A. (1982) *Classical Mechanics*. New York: John Wiley & Sons.

Fodor, J. A., & Pylyshyn, Z. (1988) Connectionism and cognitive architecture: a critical analysis. *Cognition*, **28**, 3-71.

Forrest, S. (Ed.). (1991). *Emergent computation: self-organizing, collective, and cooperative phenomena in natural and artificial computing networks.* Cambridge MA: MIT Press.

Freeman, W. J., & Skarda, C. A. (1990) Representations: Who needs them? In J. L. McGaugh, J. L. Weinberger, & G. Lynch ed., *Brain Organization and Memory: Cells, Systems and Circuits.* New York: Guildford Press.

Gregson, R. A. M. (1993) Learning in the context of nonlinear psychophysics: The Gamma Zak Embedding. *British Journal of Mathematical and Statistical Psychology*, **46**, 31-48.

Griffiths, J. B. (1985) *The Theory of Classical Dynamics.* Cambridge: Cambridge University Press.

Grossberg, S., & Gutowski, W. E. (1987) Neural dynamics of decision making under risk: affective balance and cognitive-emotional interactions. *Psychological Review*, **94**, 303-318.

Grzegorczyk, A. (1957) On the definitions of computable real continuous functions. *Fundamenta Mathematica*, **44**, 61-71.

Harnad, S. (1990) The symbol grounding problem. *Physica D*, **42**, 335-346.

Haugeland, J. (1985) *Artificial Intelligence: The Very Idea*. Cambridge MA: MIT Press.

Haugeland, J. (1991) Representational Genera. In W. Ramsey, S. P. Stich, & D. E. Rumelhart ed., *Philosophy and Connectionist Theory*. Hillsdale NJ: Lawrence Erlbaum Associates.

Hirsch, M. (1984) The dynamical systems approach to differential equations. *Bulletin of the American Mathematical Society*, **11**, 1-64.

Leven, S. J., & Levine, D. S. (1996) Multiattribute decision making in context: a dynamic neural network methodology. *Cognitive Science*, **20**.

Luenberger, D. G. (1979) *Introduction to Dynamic Systems: Theory, Models, and Applications.* New York: John Wiley & Sons.

Newell, A., & Simon, H. (1976) Computer science as empirical enquiry: Symbols and search. *Communications of the Association for Computing Machinery*, **19**, 113-126.

Ott, E. (1993) *Chaos in Dynamical Systems*. Cambridge: Cambridge University Press.

Port, R., & van Gelder, T. J. (1995) *Mind as Motion: Explorations in the Dynamics of Cognition*. Cambridge MA: MIT Press.

Pour-El, M., & Richards, J. (1989) *Computability in Analysis and Physics*. New York: Springer-Verlag.

Searle, J. R. (1980) Minds, brains and programs. *Behavioral and Brain Sciences*, **3**, 417-458.

Siegelmann, H. T., & Sontag, E. D. (1994) Analog Computation via Neural Networks. *Theoretical Computer Science*, **131**, 331-360.

Skarda, C. A., & Freeman, W. J. (1987) How brains make chaos to make sense of the world. *Behavior and Brain Sciences*, **10**, 161-195.

van Gelder, T. J. (1995) What might cognition be, if not computation? *Journal of Philosophy*, **91**, 345-381.

van Gelder, T. J. (forthcoming) The dynamical hypothesis in cognitive science. *Behavioral and Brain Sciences.*

Zak, M. (1990) Creative dynamics approach to neural intelligence. *Biological Cybernetics*, **64**, 15-23.

Captions

Table 1.

Some examples of common definitions of the term “dynamical system” from outside cognitive science, arranged roughly in order, from older narrower definitions to more recent wider ones.

Figure 1.

The basic structure of classical dynamical modeling. The natural phenomenon to be understood—the target—is assumed to be the behaviour of a concrete dynamical system, whose exact nature generally remains unknown. Scientists gain a measure of understanding of the target system by constructing an abstract system to serve as a model, and then comparing the behavior of the model with the observed behavior of the target system. A computer—typically a digital computer—is used as a tool in exploring the properties of the abstract model, and thereby indirectly those of the target.

Figure 2.

The basic structure of computational modeling in cognitive science. The target phenomenon is assumed to be the behaviour of some concrete digital computer. The exact nature of this computer may remain unknown. Scientists gain some measure of understanding of the target system by comparing it with a model system, an abstract digital computer which is similar in relevant respects. This abstract computer may be realized by a concrete digital computer, a concrete instantiation of the abstract model. The concrete computer is standardly implemented by means of a general purpose electronic digital computer.