THE JOURNAL OF PHILOSOPHY

VOLUME XCI, NO. 7, JULY 1995

WHAT MIGHT COGNITION BE, IF NOT COMPUTATION?*

What is cognition? Contemporary orthodoxy maintains that it is computation: the mind is a special kind of computer, and cognitive processes are the rule-governed manipulation of internal symbolic representations. This broad idea has dominated the philosophy and the rhetoric of cognitive science—and even, to a large extent, its practice—ever since the field emerged from the postwar cybernetic melee. It has provided the general framework for much of the most well-developed and insightful research into the nature of mental operation. Yet, over the last decade or more, the computational vision has lost much of its lustre. Although work within it continues apace, a variety of difficulties and limitations have become increasingly apparent, and researchers across cognitive science and related disciplines have been casting around for other ways to understand cognitive processes. Partly as a result, there are now many research programs which, one way or another, stand opposed to the traditional computational approach; these include connectionism, neurocomputational approaches, ecological psychology, situated robotics, synergetics, and artificial life.

These approaches appear to offer a variety of differing and even conflicting conceptions of the nature of cognition. It is therefore an appropriate time to step back and reconsider the question: What general arguments are there in favor of the idea that cognitive processes must be specifically *computational* in nature? In order prop-

* Criticism and advice from numerous people helped improve this paper, but special acknowledgement is due to Robert Port, John Haugeland, and James Townsend. Audiences at the University of Illinois/Chicago, the New Mexico State University, Indiana University, the Australian National University, the University of New South Wales, Princeton University, Lehigh University, and the University of Skövde were suitably and helpfully critical of earlier versions.

0022-362X/95/9207/345-81

erly to address this question, however, we must first address another: What are the alternatives? What *could* cognition be, if it were *not* computation of some form or other?

There are at least two reasons why this second question is important. First, arguments in favor of some broad hypothesis are rarely, if ever, completely general. They tend to be arguments not for *A* alone, but rather in favor of *A* as opposed to *B*, and such arguments often fail to support *A* as opposed to *C*. For example, one of the most powerful early considerations raised in favor of the computational conception of cognition was the idea that intelligent behavior requires sophisticated internal representations. While this clearly supported the computational conception against a behaviorism which eschewed such resources, however, it was no use against a connectionism which helped itself to internal representations, though rather different in kind than the standard symbolic variety.

The second reason we need to ask what alternatives there may be is that one of the most influential arguments in favor of the computational view is the claim that there is simply no alternative. This is sometimes known as the *"what else could it be?"* argument.[1] As Allen Newell[2] recently put it:

> ...although a small chance exists that we will see a new paradigm emerge for mind, it seems unlikely to me. Basically, there do not seem to be any viable alternatives. This position is not surprising. In lots of sciences we end up where there are no major alternatives around to the particular theories we have. Then, all the interesting kinds of scientific action occur inside the major view. It seems to me that we are getting rather close to that situation with respect to the computational theory of mind (*ibid.*, p. 56).

This paper describes a viable alternative. Rather than computers, cognitive systems may be dynamical systems; rather than computation, cognitive processes may be state-space evolution within these very different kinds of systems. It thus disarms the "what else could it be?" argument, and advances the broader project of evaluating competing hypotheses concerning the nature of cognition. Note that achieving these goals does not require decisively establishing that the dynamical hypothesis is true. That would require considerably more space than is available here, and to attempt it now would be hopelessly premature anyway. All that must be done is to describe

[1] This title may have been first used in print by John Haugeland in "The Nature and Plausibility of Cognitivism," *Behavioral and Brain Sciences*, I (1978): 215–26.

[2] "Are There Alternatives?" in W. Sieg, ed., *Acting and Reflecting* (Boston: Kluwer, 1990).

and motivate the dynamical conception sufficiently to show that it does in fact amount to an alternative conception of cognition, and one which is currently viable, as far as we can now tell.

A fruitful way to present the dynamical conception is to begin with an unusual detour, via the early industrial revolution in England, circa 1788.

I. THE GOVERNING PROBLEM

A central engineering challenge for the industrial revolution was to find a source of power that was reliable, smooth, and uniform. In the latter half of the eighteenth century, this had become the problem of translating the oscillating action of the steam piston into the rotating motion of a flywheel. In one of history's most significant technological achievements, Scottish engineer James Watt designed and patented a gearing system for a rotative engine. Steam power was no longer limited to pumping; it could be applied to any machinery that could be driven by a flywheel. The cotton industry was particularly eager to replace its horses and water wheels with the new engines. High-quality spinning and weaving required, however, that the source of power be highly uniform, that is, there should be little or no variation in the speed of revolution of the main driving flywheel. This is a problem, since the speed of the flywheel is affected both by the pressure of the steam from the boilers, and by the total workload being placed on the engine, and these are constantly fluctuating.

It was clear enough how the speed of the flywheel had to be regulated. In the pipe carrying steam from the boiler to the piston there was a throttle valve. The pressure in the piston, and so the speed of the wheel, could be adjusted by turning this valve. To keep engine speed uniform, the throttle valve would have to be turned, at just the right time and by just the right amount, to cope with changes in boiler pressure and workload. How was this to be done? The most obvious solution was to employ a human mechanic to turn the valve as necessary. This had a number of drawbacks, however: mechanics required wages, and were often unable to react sufficiently swiftly and accurately. The industrial revolution thus confronted a second engineering challenge: design a device which can automatically adjust the throttle valve so as to maintain uniform speed of the flywheel despite changes in steam pressure or workload. Such a device is known as a *governor.*

Difficult engineering problems are often best approached by breaking the overall task down into simpler subtasks, continuing the process of decomposition until one can see how to construct devices that can directly implement the various component tasks. In the case of the governing problem, the relevant decomposition seems clear. A change need only be made to the throttle valve if the flywheel is not currently running at the correct speed. Therefore, the first subtask must be to measure the speed of the wheel, and the second subtask must be to calculate whether there is any discrepancy between the desired speed and the actual speed. If there is no discrepancy, no change is needed, for the moment at least. If there is a discrepancy, then the governor must determine by how much the throttle valve should be adjusted to bring the speed of the wheel to the desired level. This will depend, of course, on the current steam pressure, and so the governor must measure the current steam pressure and then on that basis calculate how much to adjust the valve. Finally, of course, the valve must be adjusted. This overall sequence of subtasks must be carried out as often as necessary to keep the speed of the wheel sufficiently close to the desired speed.

A device that can solve the governing problem would have to carry out these various subtasks repeatedly in the correct order, and so we can think of it as obeying the following algorithm:

1. Measure the speed of the flywheel.
2. Compare the actual speed against the desired speed.
3. If there is no discrepancy, return to step 1. Otherwise,
   a. measure the current steam pressure;
   b. calculate the desired alteration in steam pressure;
   c. calculate the necessary throttle valve adjustment.
4. Make the throttle valve adjustment.

Return to step 1.

There must be some physical device capable of actually carrying out each of these subtasks, and so we can think of the governor as incorporating a tachometer (for measuring the speed of the wheel); a device for calculating the speed discrepancy; a steam pressure meter; a device for calculating the throttle valve adjustment; a throttle valve adjuster; and some kind of central executive to handle sequencing of operations. This conceptual breakdown of the components of the governor may even correspond to its actual breakdown; that is, each of these components may be implemented by a distinct, dedicated physical device. The engineering problem would then reduce to the (presumably much simpler) problem of constructing the various components and hooking them together so that the whole system functions in a coherent fashion.

Now, as obvious as this approach now seems, it was not the way the governing problem was actually solved. For one thing, it presupposes devices that can swiftly perform some quite complex calculations, and although some simple calculating devices had been invented in the seventeenth century, there was certainly nothing available in the late eighteenth century that could have met the demands of a practical governor.

The real solution, adapted by Watt from existing windmill technology, was much more direct and elegant. It consisted of a vertical spindle geared into the main flywheel so that it rotated at a speed directly dependent upon that of the flywheel itself (see figure 1). Attached to the spindle by hinges were two arms, and on the end of each arm was a metal ball. As the spindle turned, centrifugal force drove the balls outward and hence upward. By a clever arrangement, this arm motion was linked directly to the throttle valve. The result was that as the speed of the main wheel increased, the arms raised, closing the valve and restricting the flow of steam; as the speed decreased, the arms fell, opening the valve and allowing more steam to flow. The engine adopted a constant speed, maintained with extraordinary swiftness and smoothness in the presence of large fluctuations in pressure and load.

It is worth emphasizing how remarkably well the centrifugal governor actually performed its task. This device was not just an engineer-

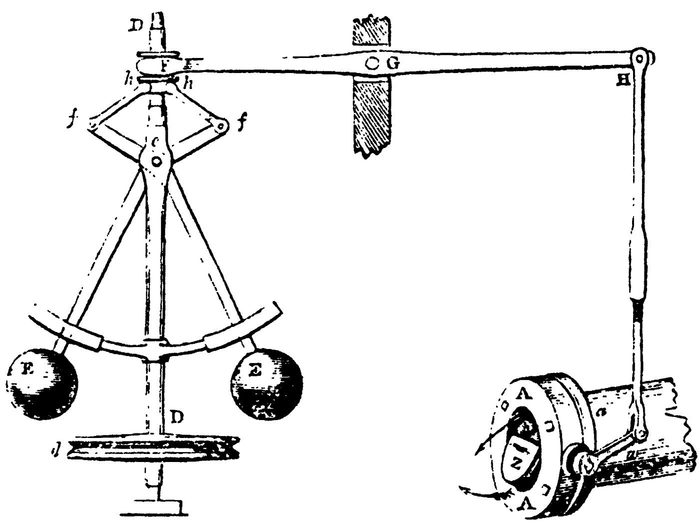

Figure 1[3]

[3] The Watt centrifugal governor for controlling the speed of a steam engine—from J. Farey, *A Treatise on the Steam Engine: Historical, Practical, and Descriptive* (London: Longman, Rees, Orme, Brown, and Green, 1827).

ing hack employed because computer technology was unavailable. *Scientific American* claimed in 1858 that an American variant of the basic centrifugal governor, "if not absolutely perfect in its action, is so nearly so, as to leave in our opinion nothing further to be desired."

But why should any of this be of any interest in the philosophy of cognitive science? The answer may become apparent as we examine a little more closely some of the differences between the two governors.

II. TWO KINDS OF GOVERNORS

The two governors described in the previous section are patently different in construction, yet they both solve the same control problem, and we can assume (for purposes of this discussion) that they both solve it sufficiently well. Does it follow that, deep down, they are really the same kind of device, despite superficial differences in construction? Or are they deeply different, despite their similarity in overt performance?

It is natural to think of the first governor as a computational device; one which, as part of its operation computes some result, namely, the desired change in throttle valve angle. Closer attention reveals that there is in fact a complex group of properties here, a group whose elements are worth teasing apart.

Perhaps the most central of the computational governor's distinctive properties is its dependence on representation. Every aspect of its operation, as outlined above, deals with representations in some manner or other. The very first thing it does is measure its environment (the engine) to obtain a symbolic representation of current engine speed. It then performs a series of operations on this and other representations, resulting in an output representation, a symbolic specification of the alteration to be made in the throttle valve; this representation then causes the valve adjusting mechanism to make the corresponding change. This is why it is appropriately described as computational (now in a somewhat narrower sense): it literally computes the desired change in throttle valve by manipulating symbols according to a schedule of rules. Those symbols, in the context of the device and its situation, have meaning, and the success of the governor in its task is owed to its symbol manipulations being in systematic accord with those meanings. The manipulations are discrete operations which necessarily occur in a determinate sequence; for example, the appropriate change in the throttle valve can only be calculated after the discrepancy between current and desired speeds has been calculated. At the highest level, the whole device operates in a cyclic fashion: it first measures (or "perceives") its environment; it then internally computes an appropriate change in throttle valve; it then effects this change ("acts" on its environment). After the change has been made and given time to affect engine speed, the governor runs through whole the cycle again...and again.... Finally, notice that the governor is homuncular in construction. Homuncularity is a special kind of breakdown of a system into parts or components, each of which is responsible for a particular subtask. Homuncular components are ones that, like departments or committees within bureaucracies, interact by communication (that is, by passing meaningful messages). Obviously, the representational and computational nature of the governor is essential to its homuncular construction: if the system as a whole did not operate by manipulating representations, it would not be possible for its components to interact by communication.

These properties—representation, computation, sequential and cyclic operation, and homuncularity—form a mutually interdependent cluster; a device with any one of them will standardly possess others. Now, the Watt centrifugal governor does not exhibit this cluster of properties as a whole, nor any one of them individually. As obvious as this may seem, it deserves a little detailed discussion and argument, since it often meets resistance, and some useful insights can be gained along the way.

Since manipulable representations lie at the heart of the computational picture, the nonrepresentational nature of the centrifugal governor is a good place to start. There is a common and initially quite attractive intuition to the effect that the angle at which the arms are swinging is a representation of the current speed of the engine, and it is because the arms are related in this way to engine speed that the governor is able to control that speed. This intuition is misleading, however; arm angle and engine speed are of course intimately related, but the relationship is not representational. There are a number of powerful arguments favoring this conclusion. They are not based on any unduly restrictive definition of the notion of representation; they go through on pretty much any reasonable characterization, based around a core idea of some state of a system which, by virtue of some general representational scheme, stands in for some further state of affairs, thereby enabling the system to behave appropriately with respect to that state of affairs.[4]

[4] This broad characterization is adapted from Haugeland, "Representational Genera," in W. Ramsey, S.P. Stich, D.E. Rumelhart, eds., *Philosophy and Connectionist Theory* (Hillsdale, NJ: Erlbaum, 1991), pp. 61–89.

A useful criterion of representation—a reliable way of telling whether a system contains them or not—is to ask whether there is any explanatory utility in describing the system in representational terms. If you really can make substantially more sense of how a system works by concretely describing various identifiable parts or aspects of it as representations in the above sense, that is the best evidence you could have that the system really does contain representations. Conversely, if describing the system as representational lets you explain nothing over and above what you could explain before, why on earth suppose it to be so? Note that very often representational descriptions do yield substantial explanatory benefits. This is certainly true for pocket calculators, and mainstream cognitive science is premised on the idea that humans and animals are like that as well. A noteworthy fact about standard explanations of how the centrifugal governor works is, however, that they never talk about representations. This was true for the informal description given above, which apparently suffices for most readers; more importantly, it has been true of the much more detailed descriptions offered by those who have actually been in the business of constructing centrifugal governors or analyzing their behavior. Thus, for example, a mechanics manual for construction of governors from the middle of last century, Maxwell's original dynamical analysis (see below), and contemporary mathematical treatments all describe the arm angle and its role in the operation of the governor in nonrepresentational terms. The reason, one might reasonably conclude, is that the governor contains no representations.

The temptation to treat the arm angle as a representation comes from the informal observation that there is some kind of correlation between arm angle and engine speed; when the engine rotates at a certain speed, the arms will swing at a given angle. Now, supposing for the moment that this is an appropriate way to describe their relationship, it would not follow that the arm angle is a representation. One of the few points of general agreement in the philosophy of cognitive science is that mere correlation does not make something a representation. Virtually everything is correlated, fortuitously or otherwise, with something else; to describe every correlation as representation is to trivialize representation. For the arm angle to count, in the context of the governing system alone, as a representation, we would have to be told what else about it justifies the claim that it is a representation.

But to talk of some kind of correlation between arm angle and engine speed is grossly inadequate, and once this is properly understood, there is simply no incentive to search for this extra ingredient.

For a start, notice that the correlation at issue only obtains when the total system has reached its stable equilibrium point, and is immediately disturbed whenever there is some sudden change in, for example, the workload on the engine. At such times, the speed of the engine quickly drops for a short period, while the angle of the arms adjusts only at the relatively slow pace dictated by gravitational acceleration. Yet, even as the arms are falling, more steam is entering the piston, and hence the device is already working; indeed, these are exactly the times when it is most crucial that the governor work effectively. Consequently, no simple correlation between arm angle and engine speed can be the basis of the operation of the governor.

The fourth and deepest reason for supposing that the centrifugal governor is not representational is that, when we fully understand the relationship between engine speed and arm angle, we see that the notion of representation is just the wrong sort of conceptual tool to apply. There is no doubt that at all times the arm angle is in some interesting way related to the speed of the engine. This is the insight which leads people to suppose that the arm angle is a representation. Yet appropriately close examination of this dependence shows exactly why the relationship cannot be one of representation. For notice that, because the arms are directly linked to the throttle valve, the angle of the arms is at all times determining the amount of steam entering the piston, and hence at all times the speed of the engine depends in some interesting way on the angle of the arms. Thus, arm angle and engine speed are at all times both determined by, and determining, each other's behavior. As we shall see below, there is nothing mysterious about this relationship; it is quite amenable to mathematical description. Yet it is much more subtle and complex than the standard concept of representation can handle, even when construed as broadly as is done here. In order to describe the relationship between arm angle and engine speed, we need a more powerful conceptual framework than mere talk of representations. That framework is the mathematical language of dynamics, and in that language, the two quantities are said to be coupled. The real problem with describing the governor as a representational device, then, is that the relation of representing—something standing in for some other state of affairs—is too simple to capture the actual interaction between the governor and the engine.

If the centrifugal governor is not representational, then it cannot be computational, at least in the specific sense that its processing cannot be a matter of the rule-governed manipulation of symbolic representations. Its noncomputational nature can also be established another way. Not only are there no representations to be manipulated, there are no distinct manipulatings that might count as computational operations. There are no discrete, identifiable steps in which one representation gets transformed into another. Rather, the system's entire operation is smooth and continuous; there is no possibility of nonarbitrarily dividing its changes over time into distinct manipulatings, and no point in trying to do so. From this, it follows that the centrifugal governor is not sequential and not cyclic in its operation in anything like the manner of the computational governor. Since there are no distinct processing steps, there can be no sequence in which those steps occur. There is never any one operation that must occur before another one can take place. Consequently, there is nothing cyclical about its operation. The device has, to be sure, an "input" end (where the spindle is driven by the engine) and an "output" end (the connection to the throttle valve). But the centrifugal governor does not follow a cycle where it first takes a measurement, then computes a throttle valve change, then makes that adjustment, then takes a measurement, and so on. Rather, input, internal activity, and output are all happening continuously and at the very same time, much as a radio is producing music at the very same time as its antenna is receiving signals.

The fact that the centrifugal governor is not sequential or cyclic in any respect points to yet another deep difference between the two kinds of governor. There is an important sense in which time does not matter in the operation of the computational governor. There is, of course, the minimal constraint that the device must control the engine speed adequately, and so individual operations within the device must be sufficiently fast. There is also the constraint that internal operations must happen in the right sequence. Beyond these, however, there is nothing that dictates when each internal operation takes place, how long it takes to carry it out, and how long elapses between each operation. There are only pragmatic implementation considerations: which algorithms to use, what kind of hardware to use to run the algorithms, and so forth. The timing of the internal operations is thus essentially arbitrary relative to that of any wider course of events. It is as if the wheel said to the governing system: "Go away and figure out how much to change the valve to keep me spinning at 100 rpm. I don't care how you do it, how many steps you take, or how long you take over each step, as long as you report back within (say) 10 milliseconds."

In the centrifugal governor, by contrast, there is simply nothing that is temporally unconstrained in this way. There are no occurrences whose timing is arbitrary relative to the operation of the engine. All behavior in the centrifugal governor happens in the very same real time frame as change in the speed of the flywheel. We can sum up the point this way: the two kinds of governor differ fundamentally in their temporality, and the temporality of the centrifugal governor is essentially that of the engine itself.

Finally, it need hardly be labored that the centrifugal governor is not a homuncular system. It has parts, to be sure, and its overall behavior is the direct result of the organized interaction of those parts. The difference is that those parts are not modules interacting by communication; they are not like little bureaucratic agents passing representations among themselves as the system achieves the overall task.

III. CONCEPTUAL FRAMEWORKS

In the previous section, I argued that the differences in nature between the two governors run much more deeply than the obvious differences in mechanical construction. Not surprisingly, these differences in nature are reflected in the kind of conceptual tools that we must bring to bear if we wish to understand the operation of these devices. That is, the two different governors require very different conceptual frameworks in order to understand how it is that they function as governors, that is, how they manage to control their environment.

In the case of the computational governor, the behavior is captured in all relevant detail by an algorithm, and the general conceptual framework we are bringing to bear is that of mainstream computer science. Computer scientists are typically concerned with what you can achieve by stringing together, in an appropriate order, some set of basic operations: either how best to string them together to achieve some particular goal (programming, theory of algorithms), or what is achievable in principle in this manner (computation theory). So we understand the computational governor as a device capable of carrying out some set of basic operations (measurings, subtractings, etc.), and whose sophisticated overall behavior results from nothing more than the complex sequencing of these basic operations. Note that there is a direct correspondence between elements of the governor (the basic processing steps it goes through) and elements of the algorithm which describes its operation (the basic instructions).

The Watt centrifugal governor, by contrast, cannot be understood this way at all. There is nothing in that device for any algorithm to lock onto. Very different conceptual tools have always been applied to this device. The terms in which it was described above, and indeed by Watt and his peers, were straightforwardly mechanical: rotations, spindles, levers, displacements, forces. Last century, more precise and powerful descriptions became available, but these also have nothing to do with computer science. In 1868, the physicist James Clerk Maxwell[5] made a pioneering extension of the mathematical tools of dynamics to regulating and governing devices. The general approach he established has been standard ever since. Though familiar to physicists and control engineers, it is less so to most cognitive scientists and philosophers of mind, and hence is worth describing in a little detail.

The key feature of the governor's behavior is the angle at which the arms are hanging, for this angle determines how much the throttle valve is opened or closed. Therefore, in order to understand the behavior of the governor, we need to understand the basic principles governing how arm angle changes over time. Obviously, the arm angle depends on the speed of the engine; hence we need to understand change in arm angle as a function of engine speed. If we suppose for the moment that the link between the governor and the throttle valve is disconnected, then this change is given by the differential equation:

$$\frac{d^2\theta}{dt^2} = (n\omega)^2 \cos\theta \sin\theta - \frac{g}{l}\sin\theta - r\frac{d\theta}{dt}$$

where θ is the angle of arms, n is a gearing constant, ω is the speed of engine, g is a constant for gravity, l is the length of the arms, and r is a constant of friction at hinges.[6] This nonlinear, second-order differential equation tells us the instantaneous acceleration in arm angle, as a function of what the current arm angle happens to be (designated by the state variable θ), how fast arm angle is currently changing (the derivative of θ with respect to time, $d\theta/dt$) and the current engine speed (ω). In other words, the equation tells us how change in arm angle is changing, depending on the current arm angle, the way it is changing already, and the engine speed. Note that in the system defined by this equation, change over time occurs only in arm angle θ (and its derivatives). The other quantities (ω, n, g, l, and r) are assumed to stay fixed, and are called parameters. The particular values at which the parameters are fixed determine the precise shape of the change in θ. For this reason, the parameter settings are said to fix the dynamics of the system.

This differential equation is perfectly general and highly succinct: it is a way of describing how the governor behaves for any arm angle and engine speed. This generality and succinctness comes at a price, however. If we happen to know what the current arm angle is, how fast it is changing, and what the engine speed is, then from this

[5] "On Governors," *Proceedings of the Royal Society*, XVI (1868): 270-83.

[6] Edward Beltrami, *Mathematics for Dynamical Modeling* (Boston: Academic, 1987), p. 163.

equation all we can figure out is the current instantaneous acceleration. If we want to know at what angle the arms will be in a half-second, for example, we need to find a solution to the general equation—that is, an equation that tells us what values θ takes as a function of time, which satisfies the differential equation. There are any number of such solutions, corresponding to all the different behavioral trajectories that the governor might exhibit, but these solutions often have important general properties in common; thus, as long as the parameters stay within certain bounds, the arms will always eventually settle into a particular angle of equilibrium for that engine speed; that angle is known as a *point attractor.*

Thus far I have been discussing the governor without taking into account its effect on the engine, and thereby indirectly on itself. Here, the situation gets a little more complicated, but the same mathematical tools apply. Suppose we think of the steam engine itself as a dynamical system governed by a set of differential equations, one of which gives us some derivative of engine speed as a function of current engine speed and a number of other variables and parameters:

$$\frac{d^n\omega}{dt^n} = F(\omega, \ldots, \tau, \ldots)$$

One of these parameters is the current setting of the throttle valve, τ, which depends directly on the governor arm angle θ. We can thus think of θ as a parameter of the engine system, just as engine speed ω is a parameter of the governor system. (Alternatively, we can think of the governor and steam engine as comprising a single dynamical system in which both arm angle and engine speed are state variables.) This relationship, known as *coupling,* is particularly interesting and subtle. Changing a parameter of a dynamical system changes its total dynamics (that is, the way its state variables change their values depending on their current values, across the full range of values they may take). Thus, any change in engine speed, no matter how small, changes not the state of the governor directly, but rather the way the state of the governor *changes,* and any change in arm angle changes the way the state of the engine changes. Again, however, the overall system (coupled engine and governor) settles quickly into a point attractor, that is, engine speed and arm angle remain constant. Indeed, the remarkable thing about this coupled system is that under a wide variety of conditions it always settles swiftly into states at which the engine is running at a particular speed. This is of course exactly what is wanted: coupling the governor to the engine results in the engine running at a constant speed.

In this discussion, two very broad, closely related sets of conceptual resources have (in a modest way) been brought into play. The first is dynamical modeling, that branch of applied mathematics which attempts to describe change in real-world systems by describing the states of the system numerically and then writing equations that capture how these numerical states change over time. The second set of resources is dynamical systems theory, the general study of dynamical systems considered as abstract mathematical structures. Roughly speaking, dynamical modeling attempts to understand natural phenomena as the behavior of real-world realizations of abstract dynamical systems, whereas dynamical systems theory studies the abstract systems themselves. There is no sharp distinction between these two sets of resources, and for our purposes they can be lumped together under the general heading of dynamics.

IV. MORALS

This discussion of the governing task suggests a number of closely related lessons for cognitive science:

(1) Various different kinds of systems, fundamentally different in nature and requiring very different conceptual tools for their understanding, can subserve sophisticated tasks—including interacting with a changing environment—which may initially appear to demand that the system have knowledge of, and reason about, its environment. The governing problem is one simple example of such a task; it can be solved either by a computational system or by a noncomputational dynamical system, the Watt centrifugal governor.

(2) In any given case, our sense that a specific cognitive task *must* be subserved by a (generically) computational system may be due to deceptively compelling preconceptions about how systems solving complex tasks must work. Many people are oblivious to the possibility of a noncomputational, dynamical solution to the governing problem, and so all-too-readily assume that it must be solved in a computational manner. Likewise, it may be that the basically computational shape of most mainstream models of cognition results not so much from the nature of cognition itself as it does from the shape of the conceptual equipment that cognitive scientists typically bring to bear in studying cognition.

(3) Cognitive systems may in fact be *dynamical* systems, and cognition the behavior of some (noncomputational) dynamical system. Perhaps, that is, cognitive systems are more relevantly similar to the centrifugal governor than they are similar either to the computational governor, or to that more famous exemplar of the broad category of computational systems, the Turing machine.

In what follows, the first and third of these points will be elaborated in just enough detail to substantiate the basic claim of this paper, that there is in fact a currently viable alternative to the computational conception of cognition. As a first step toward doing that, however, I shall briefly describe an example of dynamical research in cognitive science, in order to provide what might seem to be no more than rank speculation with a little healthy flesh.

V. AN EXAMPLE OF DYNAMICAL RESEARCH

Consider the process of coming to make a decision between a variety of options, each of which has attractions and drawbacks. This is surely a high-level cognitive task, if anything is. Psychologists have done endless experimental studies determining how people choose, and produced many mathematical models attempting to describe and explain their choice behavior. The dominant approach in modeling stems from the classic expected-utility theory and statistical decision theory as originally developed by John von Neumann and Oskar Morgenstern. The basic idea here is that an agent makes a decision by selecting the option that has the highest expected utility, which is calculated in turn by combining some formal measure of the utility of any given possible outcome with the probability that it will eventuate if the option is chosen. Much of the work within the classical framework is mathematically elegant and provides a useful description of optimal reasoning strategies. As an account of the actual decisions people reach, however, classical utility theory is seriously flawed; human subjects typically deviate from its recommendations in a variety of ways. As a result, many theories incorporating variations on the classical core have been developed, typically relaxing certain of its standard assumptions, with varying degrees of success in matching actual human choice behavior. Nevertheless, virtually all such theories remain subject to some further drawbacks:

(1) They do not incorporate any account of the underlying motivations that give rise to the utility that an object or outcome holds at a given time.
(2) They conceive of the utilities themselves as static values, and can offer no good account of how and why they might change over time, and why preferences are often inconsistent and inconstant.
(3) They offer no serious account of the deliberation process, with its attendant vacillations, inconsistencies, and distress; and they have

nothing to say about the relationships that have been uncovered between time spent deliberating and the choices eventually made.

Curiously, these drawbacks appear to have a common theme; they all concern, one way or another, *temporal* aspects of decision making. It is worth asking whether they arise because of some deep structural feature inherent in the whole framework which conceptualizes decision-making behavior in terms of calculating expected utilities.

Notice that utility-theory based accounts of human decision making ("utility theories") are deeply akin to the computational solution to the governing task. That is, if we take such accounts as not just describing the outcome of decision-making behavior, but also as a guide to the structures and processes that underlie such behavior,[7] then there are basic structural similarities to the computational governor. Thus, utility theories are straightforwardly computational; they are based on static representations of options, utilities, probabilities, and so on, and processing is the algorithmically specifiable internal manipulation of these representations to obtain a final representation of the choice to be made. Consequently, utility theories are strictly sequential; they presuppose some initial temporal stage at which the relevant information about options, likelihoods, and so on, is acquired; a second stage in which expected utilities are calculated; and a third stage at which the choice is effected in actual behavior. And, like the computational governor, they are essentially atemporal; there are no inherent constraints on the timing of the various internal operations with respect to each other or change in the environment.

What we have, in other words, is a model of human cognition which, on one hand, instantiates the same deep structure as the computational governor, and on the other, seems structurally incapable of accounting for certain essentially temporal dimensions of decision-making behavior. At this stage, we might ask: What kind of model of decision-making behavior we would get if, rather, we took the *centrifugal* governor as a prototype? It would be a model with a relatively small number of continuous variables influencing each other in real time. It would be governed by nonlinear differential equations. And it would be a model in which the agent and the choice environment, like the governor and the engine, are tightly interlocked.

[7] See, for example, J.W. Payne, J.R. Bettman, and E.J. Johnson, "Adaptive Strategy Selection in Decision Making," *Journal of Experimental Psychology: Learning, Memory, Cognition*, XIV (1988): 534–52.

It would, in short, be rather like the *motivational oscillatory theory* (MOT) modeling framework described by mathematical psychologist James Townsend.[8] MOT enables modeling of various qualitative properties of the kind of cyclical behaviors that occur when circumstances offer the possibility of satiation of desires arising from more or less permanent motivations; an obvious example is regular eating in response to recurrent natural hunger. It is built around the idea that in such situations, your underlying motivation, transitory desires with regard to the object, distance from the object, and consumption of it are continuously evolving and affecting each other in real time; for example, if your desire for food is high and you are far from it, you will move toward it (that is, z changes), which influences your satiation and so your desire. The framework thus includes variables for the current state of motivation, satiation, preference, and action (movement), and a set of differential equations describe how these variables change over time as a function of the current state of the system.[9]

[8] See "A Neuroconnectionistic Formulation of Dynamic Decision Field Theory," in D. Vickers and P.L. Smith, eds., *Human Information Processing: Measures, Mechanisms, and Models* (Amsterdam: North Holland, 1988); and "Don't Be Fazed by PHASER: Beginning Exploration of a Cyclical Motivational System," *Behavior Research Methods, Instruments and Computers*, XXIV (1992): 219–27.

[9] The equations, with rough and partial translations into English, are:

$$\frac{dx}{dt} = M - m - c$$

(The change in motivation depends on how the current levels of motivation and of consumption compare with some standard level of motivation, M.)

$$\frac{dx}{dt} = \left[\frac{1}{z_1^2 + z_2^2 + a} + 1\right] \cdot m$$

(The change in one's *preference* for the goal will depend on current motivation and one's distance from the object of preference.)

$$\frac{dc}{dt} = (x + C - c) \cdot \left[\frac{b}{z_1^2 + z_2^2 + r}\right]$$

(The change in consumption will depend on the level of preference, the level of consumption, and the distance from the object of preference.)

$$\frac{dz_1}{dt} = -x \cdot z_1 \qquad \frac{dz_2}{dt} = -x \cdot z_2$$

(How one moves toward or away from the object depends on one's current level of preference for the object.) See "Don't Be Fazed by PHASER" for an accessible and graphic introduction to the behaviors defined by these equations.

MOT stands to utility theories in much the same relation as the centrifugal governor does to the computational governor. In MOT, cognition is not the manipulation of symbols, but rather state-space evolution in a dynamical system. MOT models produce behavior which, if one squints while looking at it, seems like decision making—after all, the agent will make the move which offers the most reward, which in this case means moving toward food if sufficiently hungry. But this is decision making without decisions, so to speak, for there never are in the model any discrete internal occurrences that one could characterize as decisions. In this approach, decision making is better thought of as the behavior of an agent under the influence of the pushes and pulls that emanate from desirable outcomes, undesirable outcomes, and internal desires and motivations; in a quasi-gravitational way, these forces act on the agent with strength varying as a function of distance.

The MOT modeling framework is a special case of a more general (and rather more complex) dynamical framework which Townsend and Jerome Busemeyer[10] call "decision field theory." That framework allows faithful modeling of a wide range of behaviors more easily recognizable as decision making as studied within the traditional research paradigm; indeed, their claim is that decision field theory "covers a broader range of phenomena in greater detail" than classical utility theories, and even goes beyond them by explaining in a natural way several important paradoxes of decision making, such as the so-called "common consequence effect" and "common ratio effect." The important point for immediate purposes, however, is that the general decision field theory works on the same fundamental dynamical principles as MOT. There is thus no question that at least certain aspects of human high-level cognitive functioning can be modeled effectively using dynamical systems of the kind that can be highlighted by reference to the centrifugal governor.

Thus far, all I have done is to use the governing problem as a means of exploring some of the deep differences between computational and noncomputational solutions to complex tasks, drawn out some suggestive implications for cognitive science, and used the Busemeyer and Townsend work to illustrate the claim that high-level cognitive processes can in fact be modeled using noncomputa-

[10] "Decision Field Theory: A Dynamic-Cognitive Approach to Decision Making in an Uncertain Environment," *Psychological Review*, C (1993): 432–59; an accessible overview is given in "Dynamic Representation of Decision Making," in R. Port and myself, eds., *Mind as Motion: Explorations in the Dynamics of Cognition* (Cambridge: MIT, 1995).

tional, dynamical systems. But these moves do not really describe an alternative to the computational conception so much as just gesture in that general direction. What we need now is a sharper characterization of the dynamical conception of cognition, and some reason to suppose that the dynamical conception really is viable as a general alternative.

VI. THREE CONCEPTIONS OF COGNITIVE SYSTEMS

At the outset of this paper, I suggested that in order properly to evaluate the computational conception of cognition, we really need to know what viable alternatives there are (if any). Moreover, ideally, we would have some understanding of what the entire range of alternatives is, for only this way can we be sure that the candidates we are entertaining are in fact the most relevant. In other words, we need to be able to see the computational conception and its alternatives as options within a common field which contains all relevant possibilities.

Fortunately, the easiest way to present a sharpened characterization of the dynamical approach is in fact to sketch a common field within which can be situated, if not every conceivable option, at least the current main contenders—the computational, connectionist, and dynamical conceptions. The common field is the "space" of all state-dependent systems. A (concrete) state-dependent system is a set of features or aspects of the world which change over time interdependently, that is, in such a way that the nature of the change in any member of the system at a given time depends on the state of the members of the system at that time.[11] The most famous example from the history of science is, of course, the solar system: the positions and momentums of the sun and various planets are constantly changing in a way that always depends, in a manner captured in the laws first laid down by Newton, on what they happen to be. Another example is the Watt centrifugal governor, as described above: its future arm angles are determined by its current arm angle (and current rate of change of arm angle) according to its differential equation. And for our purposes, another particularly important category is that of computers: systems whose states are basically configurations of symbols and whose state at time $t + 1$ is always determined according to some rule by their state at time t.

Consider two centrifugal governors that are identical in all relevant physical detail. These devices will respond in exactly the same

[11] The notion of a *state-dependent* system is a generalization of that of a *state-determined* system (see Ross Ashby, *Design for a Brain* (London: Chapman and Hall, 1952)) to allow for systems in which the relation between change and current state is stochastic rather than deterministic.

way to a given engine speed; that is, their arm angles will pass through exactly the same sequences of positions over time. These two concrete systems share an abstract structure in their behavior. This structure can be distilled out, and its general properties studied, independently of any particular mechanical device. This mathematical structure is an example of an abstract state-dependent system. Generally speaking, concrete systems belong to the real world; they exist in time, and have states that change over time. Abstract systems, on the other hand, exist only in the timeless and changeless realm of pure mathematical form. They can be regarded as having three components: a set of entities (for example, the real numbers) constituting "states"; a set (for example, the integers) corresponding to points of "time," and a rule of evolution which pairs states with times to form sequences or trajectories. Thus, even if no centrifugal governor had ever been invented, mathematicians could study the abstract state-dependent system (or rather, family of systems)

$$\left\langle R^2, R, \frac{d^2\theta}{dt^2} = (n\omega)^2 \cos\theta \sin\theta - \frac{g}{l}\sin\theta - r\frac{d\theta}{dt}\right\rangle$$

where $(\theta, d\theta/dt)$ picks out points in R^2 (two dimensional Euclidean space) and the differential equation determines sequences of such points.

Abstract state-dependent systems can be realized ("made real") by particular parts (sets of aspects) of the real, physical world, as when a particular centrifugal governor realizes the abstract system just specified. An abstract system is realized by some part of the world when we can systematically classify its states (for example, by measurement) such that the sequences of states the concrete system undergoes is found to replicate the sequences specified by the abstract model. In fact, in order to count as a system at all, any concrete object must realize some abstract system or other (but not vice versa).

Now, when cognitive scientists come to study cognitive systems, whose basic nature is a matter for empirical investigation, they often proceed by providing models. Generally speaking, a model is another entity which is either better understood already, or somehow more amenable to exploration, and which is similar in relevant respects to the explanatory target. Scientific models are either concrete objects, or—more commonly—abstract mathematical entities; very often, they can be understood as state-dependent systems. If a model is sufficiently good, then we suppose that it somehow captures the nature of the explanatory target. What does this mean? Well, if the model is an abstract state-dependent system, then we suppose that the target system realizes the abstract system, or one relevantly like it. If the model is a concrete system, then we suppose that the model and the target system are systems of the same kind, in the sense that they both realize the same abstract system (or relevantly similar systems). Thus, even when providing a concrete model, what the scientist is really interested in determining is the abstract structure in the behavior of the target system.

There is a vast range of abstract state-dependent systems. Schools of thought which differ over the nature of cognition can be seen as differing over which of these abstract systems are realized by cognitive systems; or, put differently, as differing over *where* in the range of all possible systems the best models of cognition are to be found. So we can understand everyone as agreeing that cognitive systems are state-dependent systems of some kind, but as disagreeing as to which more particular category of state-dependent systems they belong. As will be explained below, this disagreement by no means exhausts the differences between the various schools of thought. Their differing commitments as to the relevant category of systems do, however, constitute a kind of core difference, around which their other differences can be organized.

1. *The computational hypothesis.* In one of the most well-known presentations of the computational conception of cognition, Newell and Herbert Simon[12] hypothesized that "physical symbol systems contain the necessary and sufficient means for general intelligent action," where a physical symbol system is "a machine that produces through time an evolving collection of symbol structures." Bearing this in mind, as well as other well-known characterizations of essentially the same target (for example, John Haugeland's definition of computers as interpreted automatic formal sytems, and various paradigm examples of computational systems such as Turing machines, pocket calculators, and classic AI systems such as Newell and Simon's GPS, Terry Winograd's SHRDLU, and Doug Lenat's CYC)[13] we can characterize the computational subcategory of state-dependent systems as follows: (ab-

[12] "Computer Science as Empirical Inquiry: Symbols and Search," in Haugeland, ed., *Mind Design* (Cambridge: MIT, 1981): pp. 35–66, here p. 40.

[13] See Haugeland, *Artificial Intelligence: The Very Idea* (Cambridge: MIT, 1985); Newell and Simon, "GPS, A Program That Simulates Human Thought," in E.A. Feigenbaum and J. Feldman, eds., *Computers and Thought* (New York: McGraw-Hill, 1963); Terry Winograd, *Understanding Natural Language* (New York: Academic, 1972); D.B. Lenat and R.V. Guha, *Building Large Knowledge-based Systems: Representation and Inference in the CYC Project* (Reading, MA: Addison-Wesley, 1990).

stract) computational systems are abstract state-dependent systems whose states are constituted in part by configurations of symbol types, whose time set is the integers (or some equivalent set), and whose rule of evolution specifies sequences of such configurations. A concrete computational system—a computer—is any system realizing an abstract computational system. In order to realize such a system, some chunk of the actual world must realize the sequences of configurations of symbol types specified by the abstract system. This means that, at any given time, it must contain an appropriate configuration of *tokens* of the symbol types, and it must change sequentially from one such configuration to another in accordance with the rule of evolution.

For example, consider a particular abstract Turing machine, Minsky's four symbol, seven head-state universal Turing machine defined by the following machine table:[14]

	1	2	3	4	5	6	7
Y	_L1	_L1	YL3	YL4	YR5	YR6	_R7
_	_L1	YR2	HALT	YR5	YL3	AL3	YR6
1	1L2	AR2	AL3	1L7	AR5	AR6	1R7
A	1L1	YR6	1L4	1L4	1R5	1R6	_R2

This table dictates the specific symbol manipulations that take place in the machine. (Thus, the first square tells us that, if the head is currently in state 1 and the symbol in the cell over which the head is positioned is a 'Y', then change that symbol to a "_" (blank), move left, and "change" head state to state 1.) This machine constitutes the abstract state-dependent system, represented

$$< \{<s, p, h>\}, I, F >$$

where each total state of the system at a given time is itself a triple made up of a configuration of symbol types s (corresponding to the contents of the entire tape), a head position with respect to that configuration (p), and a head state (h). The rule of evolution F specifies sequences of total states of the system by specifying what the next (or successor) total state will be given the current total state; hence an appropriate time set for this system is the integers (I). F is essentially equivalent to the machine table above, though the machine table specifies local manipulations rather than transformations from one total state to another. The rule can be obtained by reformulation of the machine table; the result is simple in form but too ungainly to be

[14] See Marvin Minsky, *Computation: Finite and Infinite Machines* (Englewood Cliffs, NJ: Prentice-Hall, 1967).

worth laying out here.[15] Note that a computation, from this perspective, is a sequence of transitions from one total state of the computational system to another; or, in other words, a matter of *touring* the system's symbolic state space.

A general form of the computational hypothesis, then, is that cognitive systems such as people are computational systems in the sense just defined, and that cognition is the behavior of such systems, that is, sequences of configurations of symbols. An alternative form is that for any given cognitive process, the best model of that process will be drawn from the computational subcategory of systems.

Although, as mentioned above, their primary interest is in the abstract structure of the target phenomenon, for various reasons researchers in this approach standardly provide a concrete model: an actual computer programmed so that (hopefully) it realizes the same (or a relevantly similar) abstract computational system as is realized by the cognitive systems under study. If the concrete model appears able to perform actual cognitive tasks in much the way people do, then the hypothesis that people are such systems is supported. One reason to provide a concrete model is that the abstract systems themselves are too complex to be studied by purely analytical means. In order to determine whether the model has the right properties, the theorist lets a concrete version run from a variety of starting points (initial conditions), and observes its behavior. Another reason for providing a concrete model is that, given the complexity of the abstract systems, it is very difficult actually to discover that structure except through an iterative procedure of constructing a concrete model, testing it, making improvements, and so on.

2. *The dynamical hypothesis.* Recall that one suggestion coming out of the discussion of the centrifugal governor was that an interesting alternative to the computational conception is that cognitive systems may be *dynamical* systems. In order to characterize this position as an alternative within the current framework, we need a definition of dynamical systems as a subcategory of state-dependent systems, a definition which is as useful as possible in clarifying differences among various approaches to the study of cognition.

The centrifugal governor is a paradigm example of a dynamical system. Perhaps the most pertinent contrast between it and the computational governor is that the states through which it evolves are not configurations of symbols but rather numerically measurable

[15] See Marco Giunti, *Computers, Dynamical Systems, Phenomena and the Mind,* Ph.D. Dissertation (Indiana University, 1991).

arm angles and rates of change of arm angle. Generalizing this feature, and, of course, looking over the shoulder at other textbook examples of dynamical systems and the kind of systems that are employed by dynamicists in cognitive science, we ~~can~~ define dynamical systems as state-dependent systems whose states are numerical (in the abstract case, these will be numbers, vectors, etc.; in the concrete case, numerically measurable quantities) and whose rule of evolution specifies sequences of such numerical states.

The rule of evolution in the case of the centrifugal governor was a differential equation. In general, a differential equation is any equation involving a function and one or more of its derivatives; informally, for current purposes, it can be thought of as an equation that tells you the instantaneous rate of change of some aspect of the system as a function of the current state of other aspects of the system. Since our interest is in cognition as processes that occur in time, we assume that the function is one of time (for example, $\theta(t)$) and that any derivative involved is with respect to time (for example, $d\theta/dt$). Because differential equations involve derivatives, they presuppose continuity; hence the "time" set in an abstract dynamical system is standardly R, the real numbers. Dynamical systems governed by differential equations are a particularly interesting and important subcategory, not least because of their central role in the history of science.[16] But dynamical systems in the general sense just defined might also be governed by difference equations, which specify the state of the system at time $t+1$ in terms of its state at time t:

$$s_{t+1} = F(s_t)$$

and determine sequences of states, or trajectories, by repeated application or iteration. The "time" set for abstract systems defined by difference equations is standardly the integers. For example, one of the most-studied families of dynamical systems is that defined by the difference equation known as the logistic map.[17]

$$<R, I, x_{t+1} = ax_t(1 - x_t)>$$

where a is a parameter; each possible value of a makes the rule different and hence defines a distinct system.

A concrete dynamical system, of course, is any concrete system that realizes an abstract dynamical system. The realization relation-

[16] See M. Hirsch, "The Dynamical Systems Approach to Differential Equations," *Bulletin of the American Mathematical Society*, XI (1984): 1–64.

[17] For extensive discussion, see R.L. Devaney, *An Introduction to Chaotic Dynamical Systems* (Menlo Park, CA: Cummings, 1986).

ship here is quite different than in the computational case, however. Rather than configurations of tokens of symbol types, the concrete dynamical system is made up of quantities changing in a way that corresponds to the numerical sequences specified by the rule of evolution. This correspondence is set up by measuring the quantities, that is, by using some yardstick to assign a number to each quantity at any given point in time. For example, in the case of the centrifugal governor we set up a correspondence between the actual device and the abstract mathematical system by using the "degrees" yardstick to assign a number (for example, 45) to the angle of the arm at each point in time.

The dynamical hypothesis in cognitive science, then, is the exact counterpart to the computational hypothesis: cognitive systems such as people are *dynamical* systems in the sense just laid out, and cognition is state-space evolution in such systems. Alternatively, dynamicists are committed to the claim that the best model of any given cognitive process will turn out to be drawn from the dynamical subcategory of state-dependent systems.

As in the computational case, although the theorist's primary goal is to identify the relevant abstract structure, it is often necessary in practice to explore particular concrete models. It tends to be difficult, however, to set up and explore the behavior of a concrete dynamical system with the right properties. Fortunately, there is a convenient alternative: program (that is, physically configure) a computer (a concrete computational system) so that it produces sequences of symbol-configurations which *represent* points in the state trajectories of the abstract dynamical model under consideration. In such a situation, the computer does not itself constitute a model of the cognitive process, since it does not contain numerically measurable aspects changing over time in the way that aspects of the target system are hypothesized to be changing. That is, the computer does not realize the abstract dynamical model; rather, it *simulates* it.

3. *The connectionist hypothesis.* Broadly speaking, connectionists in cognitive science are those who try to understand cognition using connectionist models, which are typically characterized along something like the following lines:

> Connectionist models are large networks of simple parallel computing elements, each of which carries a numerical *activation value* which it computes from the values of neighboring elements in the network, using some simple numerical formula. The network elements, or *units*, influence each other's values through connections that carry a numeri-

cal strength, or *weight*. The influence of each unit i on unit j is the activation value of unit i times the strength of the connection from i to j.[18]

In order to comprehend connectionism within the current framework, we need to characterize connectionist models as a particular subcategory of state-dependent systems. It is clear from the description just given, however, that all connectionist models are dynamical systems in the sense of the previous section. If the network has n neural units, then the state of the system at any given time is just an n-dimensional vector of activation values, and the behavior of the network is a sequence of such vectors determined by the equations that update unit activation values. There are, of course, innumerable variations on this basic structure, and much connectionist work consists in exploring such variations in order to find a good model of some particular cognitive phenomenon.

Why then is connectionism not simply the same thing as the dynamical conception? There are two reasons, one discussed in this section, the other in the next. The first is that connectionist models are only a particular subcategory of the wider class of dynamical systems. The core connectionist hypothesis, that the best model of any given cognitive process will be a connectionist model, is thus best regarded as a more specific version of the wider dynamical hypothesis. There are plenty of dynamical systems that are *not* connectionist networks, and plenty of dynamicists in cognitive science who are not connectionists (for example, Busemeyer and Townsend in the work described above).

What then makes a dynamical system a connectionist system? Roughly, it should conform to the Smolensky characterization above. What this means in terms of species of dynamical state-dependent systems can be seen by examining a typical connectionist system, and noting those basic features which contrast with, for example, the centrifugal governor or the MOT model. Connectionist researchers Sven Anderson and Robert Port[19] used the following quite typical abstract connectionist dynamical system as a model of certain aspects of auditory pattern recognition:

$$\left\langle R^n,\ R,\ \frac{dy_i}{dt} = -\tau_i y_i(t) + \frac{1}{1 + e - \left(\sum_j w_{ij} y_j + I_i(t) + \theta_i\right)} \right\rangle$$

[18] Paul Smolensky, "On the Proper Treatment of Connectionism," *Behavioral and Brain Sciences*, XI (1988): 1–74, here p. 1.

[19] "A Network Model of Auditory Pattern Recognition," *Technical Report* XI (Indiana University Cognitive Science Program, 1990).

The network had n neural units, each with a real activation value y_i. Hence its states were points in an n-dimensional space of real numbers, that is, elements of R^n; its time set was R, and its evolution equation was the differential equation given (in schema form) above, which specifies the instantaneous rate of change in each y_i as a function of its current value, a decay parameter (τ_i), the activation of other units (y_j), the connection weights (w_{ij}), any external input (I_i), and a threshold or bias term (θ_ι). For current purposes it is not necessary fully to understand this "simple numerical formula" or the behavior of the system as a whole. Of significance here are three closely related properties of connectionist systems that it illustrates. Connectionist systems are typically:

High-dimensional: connectionist networks standardly contain tens, or even hundreds or more, of neural units, each corresponding to a dimension of the state space. This makes them considerably larger in dimension than systems found in many other standard dynamical investigations in cognitive science, other sciences such as physics, and pure mathematics.

Homogeneous: connectionist networks are homogenous in the sense that they are made up of units that all have basically the same form; or, as Randy Beer has put the point, which are just parametric variations on a common theme. Thus, in the system above, a single equation schema suffices to describe the behavior of every unit in the network, with just the particular parameter values being specific to each unit.

"Neural": connectionist systems are made up of units which are connected with others and which adjust their activation as a function of their total input, that is, of the summed weighted activations of other units. This structural property is reflected in the form of the evolution equations for connectionist models. Thus, the connectionist equation schema above includes the term $\Sigma w_{ij} y_j$ which stands for the summed input to a unit. The defining equations of connectionist systems always include a term of this general kind.

None of these properties obtains in the case of the centrifugal governor, nor in the case of the MOT model described above; both, therefore, count as good examples of nonconnectionist dynamical systems.

4. *Hypotheses and worldviews.* Thus far, the differences between the computationalist, dynamicist, and connectionist conceptions of cognition have been described simply in terms of differing commitments as to where in the space of state-dependent systems the best models of cognition are likely to be found. Yet each of these ap-

proaches is much more richly textured than this implies; they can and should be compared and contrasted in other ways as well.

At this point, the discussion of schools of thought in cognitive science connects with the earlier discussion of the governing problem. Recall that one suggestion emerging there was that cognitive systems may in fact be more similar to the centrifugal governor than to the computational governor. Recall also that the two kinds of governor were found to contrast at two distinct "levels"—that of basic properties (representation, computation, cyclic, etc.) and that of relevant conceptual framework; and that there was a kind of natural fit between these levels. It turns out that this fit is really three-way: if you have a computational state-dependent system, it naturally implements a system that is representational, sequential, cyclic, homuncular, and so on, and the most appropriate conceptual framework to bring to bear on a system that is computational at both these levels is, of course, that of computer science and mainstream computational cognitive science. Computationalists in cognitive science do not merely select models from a particular region of the space of abstract state-dependent systems; they also make strong presuppositions about the basic overall structure of cognitive systems and they use corresponding tools in thinking about how cognitive systems work.

In other words, taking cognitive systems to be state-dependent systems that proceed from one configuration of symbols to the next is part and parcel of a general vision of the nature of cognitive systems. For computationalists, the cognitive system is basically the brain, which is a kind of control unit located inside a body which in turn is located in an external environment. The cognitive system interacts with the outside world via its more direct interaction with the body. Interaction with the environment is handled by sensory and motor transducers, whose function is to translate between the purely physical events in the body and the environment and the symbolic states that are the medium of cognitive processing. The sense organs convert physical stimulation into elementary symbolic representations of events in the body and in the environment, and the motor system converts symbolic specifications of actions into movements of the muscles. Cognitive episodes take place in a cyclic and sequential fashion; first there is sensory input to the cognitive system, then the cognitive system algorithmically manipulates symbols, coming up with an output which then causes movement of the body; then the whole cycle then begins again. Internally, the cognitive system has a modular, hierarchical construction; at the highest level, there are modules corresponding to vision, language, planning, and so on, and each of these modules breaks down into simpler modules for more elementary tasks. Each module replicates in basic structure the cognitive system as a whole; thus, they take symbolic representations as inputs, algorithmically manipulate those representations, and deliver a symbolic specification as output. Note that because the cognitive system traffics only in symbolic representations, the human body and the physical environment can be dropped from consideration; it is possible to study the cognitive system as an autonomous, bodiless, and worldless system whose function is to transform input representations into output representations.

In short, in the computational vision, cognitive systems are the computational governor writ large. Of course, there are innumerable variants on the basic computational picture; any one might diverge from the standard picture in some respects, but still remain generically computational in nature (for example, symbolic models that utilize some measure of parallel processing).

The dynamical conception of cognition likewise involves interdependent commitments at three distinct levels, but stands opposed to the computational conception in almost every respect. The core dynamical hypothesis—that the best models of any given cognitive process will specify sequences, not of configurations of symbol types, but rather of numerical states—goes hand in hand with a conception of cognitive systems not as devices that transform symbolic inputs into symbolic outputs but rather as complexes of continuous, simultaneous, and mutually determining change, for which the tools of dynamical modeling and dynamical systems theory are most appropriate. In this vision, the cognitive system is not just the encapsulated brain; rather, since the nervous system, body, and environment are all constantly changing and simultaneously influencing each other, the true cognitive system is a single unified system embracing all three. The cognitive system does not interact with the body and the external world by means of the occasional static symbolic inputs and outputs; rather, interaction between the inner and the outer is best thought of as a matter of coupling, such that both sets of processes continually influencing each other's direction of change. At the level at which the mechanisms are best described, cognitive processing is not sequential and cyclic, for all aspects of the cognitive system are undergoing change all the time. Any sequential character in cognitive performance is the high-level, overall trajectory of change in a system whose rules of evolution specify not sequential change but rather simultaneous mutual coevolution.

Where does connectionism fit into all this? Perched somewhere in the middle. Recall that connectionist models are dynamical systems, but that there are reasons not simply to assimilate the connectionist and dynamical conceptions. The first was that connectionist models are really a quite specific kind of dynamical system. What we can now add is that although many connectionists are thoroughly dynamical in their general vision of the nature of cognitive systems, many others attempt to combine their connectionist dynamical substrates with an overall conception of the nature of cognitive systems which owes more to the computational worldview. Thus, consider "good old fashioned connectionism": standard, layered-network back-propagation connectionism of the kind that became fashionable with the well-known 1986 volumes. A classic exemplar is David Rumelhart and James McClelland's[20] past-tense learning model. In this kind of work, underlying systems that are basically dynamical in nature are configured so as sequentially to transform static input representations into output representations. They retain much of the basic structure of the computational picture, changing some ingredients (in particular, the nature of the representations) but retaining others. Connectionism of this kind can be regarded as having taken up a half-way house between the computational and dynamical conceptions, combining ingredients from both in what may well turn out to be an unstable mixture. If this is right, we should expect as time goes on that such connectionist models will increasingly give way either to implementations of generically computational conceptions of cognition, or to models that are more thoroughly dynamical.

VII. IS THE DYNAMICAL CONCEPTION VIABLE?

In order soundly to refute the "what else could it be?" argument, a proposed alternative must be viable, that is, plausible enough that it is reasonably deemed an open empirical question whether the orthodox approach, or the alternative, is the more correct.

One measure of the viability of an approach is whether valuable research can be carried out within its terms. On this measure, the dynamical approach is certainly in good health. Dynamical theories and models have been or are being developed of a very wide range of aspects of cognitive functioning, from (so-called) low-level or peripheral aspects such as brain function, perception, and motor control, to (so-called) central or higher aspects such as language and

[20] "On Learning the Past Tenses of English Verbs," in McClelland and Rumelhart, eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume II: Psychological and Biological Models* (Cambridge: MIT, 1986), pp. 216–68.

decision making, and through to related areas such as psychiatry and social psychology. As already mentioned, a good deal of connectionist work falls under the dynamical banner, and this work alone would qualify the dynamical approach as worth taking seriously. But there are nonconnectionist dynamical models of numerous aspects of cognition, and their ranks are swelling. In a number of fields under the broader umbrella of cognitive science, dynamics provides the dominant formal framework within which particular theories and models are developed: these include neural modeling, autonomous agent ("animat") research, ecological psychology, and, increasingly, developmental psychology.[21]

Of course, it is quite possible that a research program is flourishing, and yet there be deep reasons why it will eventually prove inadequate, either in general or with respect to particular aspects of cognition. (Consider behaviorism in its hey-day, for example.) In evaluating the plausibility of an alternative, we should also consider whether there are known *general* considerations that either strongly support—or, perhaps more importantly, stand opposed to—that approach.

Many considerations have been raised in favor of the computational conception of cognition, and, given the deep differences between the approaches, each might appear to constitute an argument against the dynamical alternative. It is not possible adequately to address all (or even any) such arguments here, but I shall briefly comment on two of the most powerful, in order to reveal not the weakness but rather something of the potential of the dynamical approach.

Cognition is distinguished from other kinds of complex natural processes (such as thunderstorms, subatomic processes, etc.) by at

[21] Rather than cite individual examples, I merely list here some overviews or collections that the interested reader can use as a bridge into the extensive realm of dynamical research on cognition. A broad sampling of current research is contained in *Mind as Motion: Explorations in the Dynamics of Cognition*; this book contains guides to a much larger literature. An excellent illustration of the power and scope of dynamical research, in a neural network guise, is S. Grossberg, ed., *Neural Networks and Natural Intelligence* (Cambridge: MIT, 1988). R. Serra and G. Zanarini, *Complex Systems and Cognitive Processes* (Berlin: Springer, 1990) presents an overview of a variety of dynamical systems approaches in artificial intelligence research. For the role of dynamics in developmental psychology, see Esther Thelen and Linda Smith, *A Dynamics Systems Approach to the Development of Cognition and Action* (Cambridge: MIT, 1993) and *Dynamic Systems in Development: Applications* (Cambridge: MIT, 1993). Hermann Haken, *Synergetic Computers and Cognition: A Top-down Approach to Neural Nets* (Berlin: Springer, 1991) provides an introduction and overview to the "synergetic" form of the dynamical approach.

least two deep features: on one hand, a dependence on knowledge; and distinctive kinds of complexity, as manifested most clearly in the structural complexity of natural languages. One challenge for cognitive scientists is to understand how a physical system might exhibit these features.

The usual approach to explaining the dependence on knowledge is to suppose that the system contains internal structures that represent that knowledge. Further, the most powerful known way of doing this is to use symbolic representations, manipulated by some computational system. Insofar as the dynamical approach abjures representation completely, or offers some less powerful representational substitute, it may seem doomed.

While the centrifugal governor is clearly a nonrepresentational dynamical system, and while it was argued above that representation figures in a natural cluster of deep features that are jointly characteristic of computational models, in fact there is nothing preventing dynamical systems from incorporating some form of representation; indeed, an exciting feature of the dynamical approach is that it offers opportunities for dramatically reconceiving the nature of representation in cognitive systems, even within a broadly noncomputational framework. A common strategy in dynamical modeling is to assign representational significance to some or all of the state variables or parameters (for example, see the Townsend and Busemeyer decision field theory model described above, or consider a connectionist network in which units stand for features of the domain). While representations of this kind may be exactly what is needed for some cognitive modeling purposes, they do not have the kind of combinatorial structure that is often thought necessary for other aspects of high-level cognition. Within the conceptual repertoire of dynamics, however, there is a vast range of entities and structures that might be harnessed into representational roles; individual state variables and parameters are merely the simplest of them. For example, it is known how to construct representational schemes in which complex contents (such as linguistic structures) are assigned in a recursive manner to points in the state space of a dynamical system, such that the representations form a fractal structure of potentially infinite depth, and such that the behavior of the system can be seen as transforming representations in ways that respect the represented structure.[22] Yet even these methods are doing little more than dipping a toe into the pool of possibilities. For ex-

[22] See, for example, Jordan Pollack, "Recursive Distributed Representations," *Artificial Intelligence*, XLVI (1990): 77–105.

ample, representations can be trajectories or attractors of various kinds, or even such exotica as transformations of attractor arrangements as a system's control parameters change.[23] Dynamicists are actively exploring how these and other representational possibilities might be incorporated into cognitive models, without buying the rest of the computational worldview. Consequently, while the dynamical approach is certainly a long way from having actual solutions to most concrete problems of knowledge representation, it clearly holds sufficient promise to maintain its current viability as an alternative.

What, then, about arguments that are based on the distinctive complexity of human cognition? Perhaps the most common, and probably the most persuasive argument of this kind focuses on the complexity of sentences of natural language. It begins from the observation that any proficient language user can understand and produce an effectively unbounded number of distinct sentences, and proceeds to note that these sentences can manifest phenomena such as repeated embedding and dependencies over arbitrarily long distances. If we attempt to describe languages with this kind of complexity by means of a grammar (a finite set of rules for combining a finite set of primitive elements into complex structures), we find they can only be compactly specified by grammars more powerful than so-called "regular" or "phrase-structure" grammars. If we then ask what kind of computational device is capable of following the rules of these grammars to recognize or produce such sentences, the answer is that they can only be implemented on machines more powerful than finite-state machines, such as push-down automata or linear-bounded automata. Therefore, human cognitive systems must be one of these more powerful computational systems.

A crucial question, then, is whether there is reason to believe that dynamical systems, with their numerical states and rules of evolution defined over them, are capable of exhibiting this order of complexity in behavior. The investigation of the "computational" power of dynamical systems, especially in the form of neural networks, is a relatively new topic, but there is already a sizable literature and results available indicate a positive answer. For example, J. P. Crutchfield and K. Young[24] have studied the complexity of the behavior in cer-

[23] See, for example, Jean Petitot, "Morphodynamics and Attractor Syntax," in *Mind as Motion: Explorations in the Dynamics of Cognition.*

[24] See J.P. Crutchfield and K. Young, "Computation at the Onset of Chaos," in W.H. Zurek, ed., *Complexity, Entropy, and the Physics of Information, SFI Studies in the Sciences of Complexity, Volume VIII* (Reading, MA: Addison-Wesley, 1990).

tain nonlinear dynamical systems "at the edge of chaos" (that is, at settings of parameters close to those settings which would produce genuinely chaotic behavior). If passing through a particular region of the state space is counted as producing a symbol, then allowing the system to run produces a sequence of symbols. It turns out that the complexity of these sequences is such that describing them requires an indexed context-free grammar. This means that the system is producing behavior of the same broad order of complexity as many believe natural language to possess.

Similarly, Jordan Pollack[25] has studied the ability of connectionist dynamical systems to recognize languages (that is, to indicate whether or not any given sequence belongs to the language). In his networks, the system bounces around its numerical state space under the influence of successive inputs corresponding to symbols in the sequence to be recognized. A well-formed sequence is regarded as successfully recognized if the system ends up in a particular region after exposure to the whole sentence, while ending up in some other region for non-well-formed sequences. Pollack (among others) has found that there are networks that can recognize nonregular languages, and in fact can learn to have this ability, via a novel form of induction in language learning, involving bifurcations in system dynamics which occur as the weights in the network gradually change.

More generally, it is clear that nonlinear dynamical systems can not only match but exceed the complexity of behavior of standard computational systems such as Turing machines.[26] Of course, this alone by no means establishes that cognitive systems are, or are more likely to be, dynamical systems than computational systems. It does establish that the dynamical approach is not automatically ruled out by these kinds of complexity considerations. What kind of system humans in fact are is therefore a question only to be resolved by means of patient and detailed modeling.

So much for defenses of viability. What positive reasons are there to think that the dynamical approach is actually on the right track? Again, space does not allow serious treatment of these arguments, but some are at least worth mentioning. In practice, an important part of the appeal of the dynamical approach is that it brings to the study of cognition tools that have proved so extraordinarily success-

[25] "The Induction of Dynamical Recognizers," *Machine Learning*, VII (1991): 227–52.

[26] See, for example, Hava Siegelmann and Eduardo Sontag, "Analog Computation via Neural Networks," *Theoretical Computer Science*, CXXX, 1 (1994): 331–60.

ful in so many other areas of science. But what is there about *cognition*, in particular, which suggests that it will be best understood dynamically?

One central fact about natural cognitive processes is that they always happen *in time*, which means not merely that, like any physical process including ordinary digital computation, they occupy some extent of actual time, but that details of *timing* (durations, rates, rhythms, etc.) are critical to a system that operates in a real body and environment. As we saw above, dynamics is all about describing how processes happen in time, while computational models are inherently limited in this respect. Cognition also has other general features for which a dynamical approach appears particularly well-suited. For example, it is a kind of complex behavioral organization that is emergent from the local interactions of very large numbers of (relatively) simple and homogenous elements. It is pervaded by both continuous and discrete forms of change. At every level, it involves multiple, simultaneous, interacting processes. Dynamics is a natural framework for developing theories that account for such features. Further, that within which cognition takes place (the brain, the body, and the environment) demand dynamical tools in their description. A dynamical account of cognition promises to minimize difficulties in understanding how cognitive systems are real biological systems in constant, intimate dependence on, or interaction with, their surrounds.[27]

A final way to underpin the viability of the dynamical conception is to place it and the computational conception in broad historical perspective. Computationalism, as cognitive science orthodoxy, amounts to a sophisticated instantiation of the basic outlines of a generically Cartesian picture of the nature of mind. Conversely, the prior grip that this Cartesian picture has on how most people think about mind and cognition makes the computational conception intuitively attractive to many people. This would be unobjectionable if the Cartesian conception was basically sound. But the upshot of philosophical evaluation of the Cartesian framework over the last three centuries, and especially this century, is that it seriously misconceives mind and its place in nature. Cognitive scientists tend to suppose that the primary respect in which Descartes was wrong about mind was in subscribing to an interactionist dualism, that is, that doctrine that mind and body are two distinct substances that

[27] For more detailed treatment of these and other arguments, see Port and my "It's about Time: An Overview of the Dynamical Approach to Cognition," in *Mind as Motion: Explorations in the Dynamics of Cognition.*

causally interact with one another. Already by the eighteenth century, however, the inadequacy of this particular aspect of Cartesianism had been repeatedly exposed, and thoroughgoing brain-based materialisms had been espoused by philosophers such as Thomas Hobbes and Julien Offray de La Mettrie. Some of the greatest achievements of twentieth-century philosophy of mind have been the exposing of various other, more subtle, pervasive, and pernicious epistemological and ontological misconceptions inherent in the Cartesian picture. These misconceptions are very often retained even when substance dualism is rejected in favor of some brain-based materialism, such as functionalism in its various guises.

For current purposes, one of the most important anti-Cartesian movements is the one spearheaded by Gilbert Ryle in Anglo-American philosophy and Martin Heidegger in "continental" philosophy.[28] Its target has been the generically Cartesian idea that mind is an inner realm of representations and processes, and that mind conceived this way is the causal underpinning of our intelligent behavior. This movement comprises at least three major components, all intimately interrelated. The first is a relocating of mind. The Cartesian tradition is mistaken in supposing that mind is an inner entity of any kind, whether mind-stuff, brain states, or whatever. Ontologically, mind is much more a matter of what we *do* within environmental and social possibilities and bounds. Twentieth-century anti-Cartesianism thus draws much of mind out, and in particular outside the skull. The second component is a reconceiving of our fundamental relationship to the world around us. In the Cartesian framework, the basic stance of mind toward the world is one of representing and thinking about it, with occasional, peripheral, causal interaction via perception and action. It has been known since Bishop Berkeley that this framework had fundamental epistemological problems. It has been a more recent achievement to show that escaping these epistemological problems means reconceiving the human agent as essentially embedded in, and skillfully coping with, a changing world; and that representing and thinking about the world is secondary to and dependent upon such embeddedness.[29] The third component is an attack on the supposition that the kind of behaviors we exhibit (such that we are embedded in our world and can

[28] See Ryle, *The Concept of Mind* (Chicago: University Press, 1984); Heidegger, *Being and Time*, John Macquarrie and Edward Robinson, trans. (New York: Harper, 1962); and Hubert Dreyfus, *Being-in-the-World: A Commentary on Heidegger's* Being and Time, *Division 1* (Cambridge: MIT, 1991).

[29] See Charles Guignon, *Heidegger and the Problem of Knowledge* (Indianapolis: Hackett, 1983).

be said to have minds) could ever be causally explained utilizing only the generically Cartesian resources of representations, rules, procedures, algorithms, and so on. A fundamental Cartesian mistake is, as Ryle variously put it, to suppose that practice is accounted for by theory; that knowledge how is explained in terms of knowledge that; or that skill is a matter of thought. That is, not only is mind not to be found wholly inside the skull; cognition, the inner causal underpinning of mind, is not to be explained in terms of the basic entities of the Cartesian conception of mind.

My concern here is not to substantiate these claims or the post-Cartesian conception of the person to which they point;[30] it is simply to make the computational conception of cognition seem less than inevitable by pointing out that serious doubt has been cast upon the philosophical framework in which it is embedded. Orthodox computational cognitive science has absorbed some of the important lessons of seventeenth-century reactions to Cartesianism, but so far has remained largely oblivious to the more radical twentieth-century critiques. Conversely, if we begin with a thoroughly post-Cartesian approach, the dynamical account of cognition will, in many ways, be immediately attractive. The post-Cartesian conception rejects the model of mind as an atemporal representer and, like the dynamical approach to cognition, emphasizes instead the ongoing, real-time interaction of the situated agent with a changing world. The post-Cartesian agent is essentially temporal, since its most basic relationship to the world is one of skillful coping; the dynamical framework is a therefore natural choice since it builds time in right from the very start. The post-Cartesian agent manages to cope with the world without necessarily representing it; a dynamical approach suggests how this might be possible by showing how the internal operation of a system interacting with an external world can be so subtle and complex as to defy description in representational terms—how, in other words, how cognition can *transcend* representation. In short, from the philosophical perspective that has managed to overcome the deep structures of the Cartesian world view, the dynamical approach looks distinctly appealing; the Watt governor is preferable to the Turing machine as a landmark for models of cognition.

TIM VAN GELDER

University of Melbourne

[30] Dreyfus, *What Computers* Still *Can't Do: A Critique of Artificial Reason* (Cambridge: MIT, 1992) is excellent in this regard.