AISB Journal **1**(3), © SSAISB, 2003
http://www.aisb.org.uk

Intelligence as Physical Computation

Sunny Bains

Intelligent and Interactive Systems, EEE Department, Imperial College London
Exhibition Road, London SW7 2BT, UK
Department of Design and Innovation, The Open University
Milton Keynes MK7 6AA, UK
sunny@sunnybains.com

Abstract

In this paper, we describe a model of physical computation that allows us to understand how embodied intelligent agents can simultaneously be considered to be objects operating under the laws of nature (or physics) and information processing devices. Using this type of analysis has two major advantages. First, it allows us to make concrete those issues relating to the relative merits of using analogue values or symbolic representations such as numbers: our analysis allows this longstanding point of contention in artificial intelligence to be transformed from a question of philosophy to one of physics. Second, it gives us a framework that should eventually allow us to produce design rules that will enable our artificially-intelligent (physical) agents to make better use of influences from the physical environment. As should be expected, for well-defined tasks, the results of this analysis are identical to Shannon's information theory. However, for large, multi-functional systems with ill-defined roles, the new model provides a novel way of thinking about the complementarity of hardware and software.

1 Introduction

In this paper, we take a fresh look at what intelligence is, where it fundamentally comes from, and how this affects the way we think about building artificially-intelligent agents. Note that this model is exclusively directed at the goal of understanding embodied intelligence and, particularly, intelligence that is based in the natural or real (as opposed to mathematical) world. In this sense, though it applies to any real machine that can perform mathematics, the model is *least* useful for those whose goal is mathematical manipulation rather than the processing of signals from the outside world in order to produce intelligent actions.

This research has come about through a desire to bridge a gap between two existing fields. First is theoretical work that proves that analogue computation, in the form of recurrent analogue neural networks (RANN), can be super-Turing in nature (Siegelmann, 1999). Siegelmann demonstrated that, if allowed to take continuous rather than discrete weights, recurrent neural networks could perform functions that are theoretically impossible using Turing machines. The problem with this model is that it is entirely mathematical: in it, there is no interface to the real (physical) world, so assumptions she made (about continuity, for instance, or noise) have no clear way of being validated. Likewise, studies that look at how noise can degrade the abilities of such machines (Maass, 1997) do so from a theoretical rather than physical perspective.

The link between the theoretical and the physical is important because the brain looks very much like a RANN. If Siegelmann's conclusions were applied to the brain, then Roger Penrose's (1989) assertion that human-like intelligence could not be performed on Turing machines could be correct (though for the wrong reasons).

It is important, at this point, to distance the arguments made in this paper from the debate Penrose famously started concerning physics, artificial intelligence and computationalism. Using Gödelian arguments (Gödel, 1931), he pointed out that certain propositions are undecidable in Turing machines (as they are in all such mathematical systems) based on the axioms inherent in those systems. He also claimed that microtubules in the brain (Penrose, 1997) had quantum-mechanical properties that were both non-Turing-computable and potentially important to intelligence. These claims have generally been disputed from both computational and physical points of view. First, it has been argued that the specific Gödelian limitations are not, in fact, in conflict with human intellectual abilities (e.g. Laforte *et al.*, 1998). Second, the timescales related to quantum decoherence in microtubules have been shown to be so different from those relevant to the brain that classical (rather than quantum) neural behaviour has been proposed as the more appropriate model (Tegmark, 2000).

Instead, this paper is more concerned with issues such as those succinctly outlined by Dreyfus in the 1970s (Dreyfus, 1972). In his book *What computers still can't do: A critique of artificial reason*, Dreyfus explains why the formalization of a physical process is not the same as the process itself (Chapter 5). With this paper, we take a more engineering-based approach to his philosophical questions. We ask, if the behaviour of physical systems cannot be replicated using Turing machines, how can they be replicated?

Back to the technological gap that we are trying to bridge. On one side we have the work done in theoretical computer science (by researchers like Siegelmann), while on the other we have the field of neuromorphic engineering: where engineers try to structure their machines, often including the hardware, in a brain-like way. Carver Mead's analysis of the power-efficiency of analogue computation (e.g. Mead, 1989), particularly for neural networks, has been extremely important in this regard. He demonstrated that by exploiting, rather than fighting against, the intrinsic physics of electronics, analogue circuits could be orders of magnitude more efficient than their digital counterparts. Leon Chua's cellular neural network (e.g. Chua, 1998), a device that is digitally programmable but can perform complicated nonlinear operations during the analogue transient—the "switching" time to go from one stable state to the next—is an apt demonstration of Mead's point. Not only is the device far lower-power than the equivalent image-processor, but it is also up to orders of magnitudes faster (depending on the algorithms implemented). Nabil Farhat's optical implementations of biological neural models (e.g. Farhat, 1997 and Farhat and Wen, 1995) not only show the utility of analogue networks, but their potential complexity. From conceptually very simple optical and/or analogue circuits, he has obtained behavior that—within a single system—varies between periodic, m-periodic (repetition of a pattern of m beats), psuedo-periodic (qualitatively periodic, but not strictly so quantitatively), and chaotic (with a fractal set as an attractor) output.

Though these researchers never made any explicit claim that their systems were computationally superior, Siegelmann's model suggested they might be. Thus, it seemed logical to find an approach that would bridge the gap between the two. To date, Siegelmann and others (e.g. Blum *et al.*, 1989) have worked with notions of super-Turing or hyper-computation: forms of computation that can perform functions theoretically impossible with conventional Turing machines, such as functions based on non-computable algebras or that are non-recursive. In particular, they have concentrated on computation over the reals. This theoretical approach, though fruitful, does not tell us what is possible with real machines. Our new model, on the other hand, was designed with the following intentions: a) that it should have a clear interface with physics; b) that it have a clear application to artificial intelligence in the simulation-of-behaviour sense (as opposed to the solution-of-arbitrary-mathematical-problems sense); and, c) that it allow for an understanding of the differences and relationships among various different types of machine.

Note that, though the model presented here does all of the above—including providing an interface with physics (in that it can be interrogated using what we know of the physical laws)—physical interrogation is beyond the scope of this paper (but is at an advanced stage).

The paper is structured as follows. In Section 2 we discuss the Turing machine and why, whatever its comparative computational power, it is an inadequate model for the embodied intelligence task. In Section 3 we introduce the basis of the new model, showing some important variations on how it can work in Section 4. The basics of how the model can be used as an interface between the computational and physical paradigms are explained in Section 5, with a discussion of the model—with particular reference to neuromorphic engineering—in Section 6. Finally, in section 7, we will describe current avenues of this research that are beyond the scope of this paper, and future avenues that may be productive.

2 The Turing machine as a model of intelligence

When Turing wrote his original paper describing the thought experiment that became known as the Turing machine, he described it as a kind of automatic version of doing what people do when they manipulate numbers and symbols on paper in an intelligent way (Turing, 1936). The machine had an arbitrarily long tape from which it could read as many symbols (from a finite set) as needed, means to read and manipulate those symbols (the read/write head), a finite set of states where it could store information about what had gone before, and a set of rules that governed what it should do in the event of various combinations of input and state.

Critically, the input and output are explicitly constrained to be symbolic. Thus, the machine is unable to interact directly with the environment: it must always receive a set of symbolic inputs from sensors, which have to perform some type of transformation into the relevant symbol space (such as digitisation), and must produce its output as a set of control symbols that can then be used to affect the real world.

This fact is critical, because it means that there is no way that a Turing machine can perform any kind of natural behavioural intelligence—where it responds to some external force or signal by moving or changing state—it can only perform the intelligent manipulation of symbols. It is, therefore, not just wrong but essentially *meaningless* to speculate on the ability of Turing machines to be able to perform human-like intelligence tasks. In any real machine, but particularly those designed to interact with the environment, the outer shell (body, sensors, actuators) must be, by definition, entirely analogue. This is because the signals that they deal with (from the outside world) are analogous to the real physical values in question (or, more precisely, they *are* the real values). For the machine to work, at some stage after this outer-shell has been breached, an analogue-to-digital (a/d) conversion step must take place, thus allowing the digital computer to do whatever processing is required. The same, in reverse, is true for actuation.

Given this, every robot (and, to a lesser extent, every computer) is a hybrid machine:

part analogue, part digital. In the following sections, a crucial question in the design process of any machine, but particularly a machine intended to be tightly-coupled to its environment, is highlighted: the location of the boundary between these two parts. In particular, we believe the virtual interaction variation of the model gives some insight into this question though, as yet, it falls short of providing specific design rules.

Note that the above argument is not anti-computationalist in the traditional sense. Rather, the intention here is to point out that computationalism as it is normally discussed is moot. Since all robots must be hybrid machines (and, therefore, not *necessarily* constrained by the well-known limits of Turing computation), setting up the Turing machine as the only route to artificial intelligence is, in effect, setting up a straw man. From a theoretical perspective the question is whether the hybrid machine has the computational power to do the job. From an engineering perspective, the question is how such a machine can be designed to do the job *most efficiently*.

3 The physical computational model

Here we present a mathematical model of a potentially-intelligent, embodied, adaptive, physical system: one that includes mechanisms that can be interpreted as allowing learning via experience of the environment and action based on that experience. In Section 5 we consider the elements that are affected by physical constraints (i.e. the constraints of the real physical world as opposed to specific engineering constraints), but for now we simply lay out the mathematical model.

A *system* is here defined as an identifiable collection of connected elements. A system is said to be *embodied* if it occupies a definable volume and has a collective contiguous boundary. The matter, space and energy outside the boundaries of the embodied system are collectively called the *environment*.

A *sensor* is any part of the system that can be changed by physical influences from the environment. These, which include any or all forces, fields, energy, matter, etc. that may be impinging on the system, are collectively called the *sensor input* ($x \in \mathcal{X}$), even where no explicitly-defined sensors exist. Though represented by a single variable, the sensor input may in fact consist of many different *sensor modalities* (each influenced by a different type of force or energy).

Resulting external physical changes to the embodied system, (emission of light, movement of a limb, etc.), are collectively called the *actuator output* ($y \in \mathcal{Y}$) of this function. An *actuator* is any part of the system that can change the environment. A coupled pair of sensor input and actuator output may be described as a *behaviour*.

Let us define $\mathbf{G}$ as the *intelligence function* performed by the embodied system, mapping the input to the output. Where t is time, and $\delta t > 0$ but arbitrarily small (in other words, time may be continuous—whether it is or not is a physical question that will be addressed in a future paper), we have:

$$\mathbf{G}_t(x_t) \rightarrow y_{t+\delta t}$$

It is important to note here that, for our purposes, $\mathbf{G}_t$ can only be considered to cause an immediate actuator output (change that may effect the environment) as a result of an immediate actuator input (physical influence from the environment). It cannot be considered to implement any kind of plan over time, like commanding robot arm to move through a particular trajectory. Instead, the plan is carried out through $\mathbf{G}$ itself changing with time. $\mathbf{G}$ is altered by any internal changes to the system caused by the input (flow of a current inside a wire, charging of a capacitor, shifting of weight, etc.). The *learning* *function*, $\mathbf{L_G}$, determines how $\mathbf{G}$ changes with time: $\mathbf{L_G}(\mathbf{G}_t, x_{t-\delta t}) \rightarrow \mathbf{G}_{t+\delta t}$. This process is called *adaptation*.

This is a subtle, but important, difference in the way we think about how machines work. Were the intelligence able to issue a command to be followed by an actuator over time, some controller would have to be at work in the actuator to make sure that the command was carried out. This is fine, but in our system this controller is considered to be *part* of the intelligence function.

Now, let x' be the output from the environment to the system, and y' be the system's input to the environment: the impact of the system's behaviour on its surroundings. Where $\mathbf{E}$ is the *environment reaction function* mapping the input to the output:

$$\mathbf{E}_t(y'_t) \rightarrow x'_{t+\delta t}$$

There is also an *environment learning function*, $\mathbf{L_E}$, equivalent to $\mathbf{L_G}$, that determines how $\mathbf{E}$ changes with time.

4 Categories of physical interaction

4.1 Real interaction

The interaction between $\mathbf{E}$ and $\mathbf{G}$ may be considered to fall into one of two classes. *Real interaction* is a pure physical process in which embodied intelligence *co-evolves* with its environment: such that the two functions are dependent only on initial conditions, their governing functions ($\mathbf{L_E}$ and $\mathbf{L_G}$), their interactions with each other, and time. Thus, any adaptation of the embodied system is in direct response to its environment and nothing else (and vice versa). See Figure 1. This type of interaction requires that $x = x'$ and $y = y'$. Consequently, the domains/ranges of the two systems must be the same.

In the real-interaction scenario, as well as being a function, $\mathbf{G}$ may also be considered a description of the system's *instantaneous physical state* at the arrival of the input stimulus. It is important to note that the state is here defined as not only specific parameters that can be measured instantaneously (speed, position, etc.), but also all associated rates of change. For example, two balls—one at rest and the other falling under gravity—would not be considered to have the same intelligence function even if they were identical in all other ways. Instantaneously they might look the same, but with associated rates of change taken into account, they are clearly not.

This should also highlight how intelligence in the physical sense is different from our idea of intelligence in computers normally. With conventional computers, we consider intelligence functions on *subsets* of stimuli and represented as abstractions. This makes them implementation independent: an adder can be built in many different ways with many different materials, but intelligence is required to determine what this intelligence function actually is and to *interpret*—which we can define as weeding out the relevent from the irrelevent—the results. With physical computation, the function and implementation are one and the same.

In the real-interaction case, implemented in the physical world, various constraints may be imposed. These include the following:

- The combined values of certain physical parameters of the two systems may be conserved (e.g. conservation of energy).

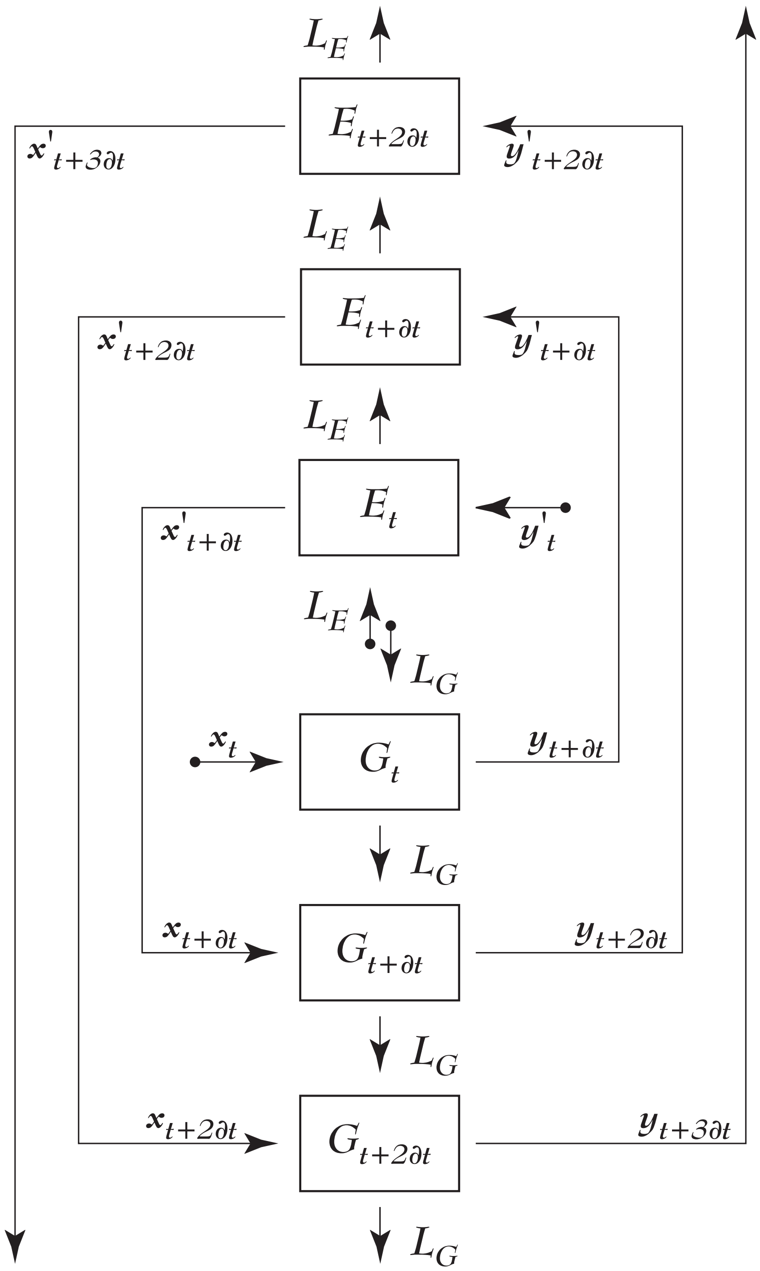

Figure 1: Here an object is evolving under physics. **G** is the function that the object performs on the inputs (x) it gets from the outside world, which determine how it will impact the outside world (y). Likewise for **E**, which is the function performed by the outside world on the input from the object (y'), producing the output (x'). These two functions are carried out in parallel. Though labelled differently, x and x' must be the same for real interaction to take place, and likewise for y and y': thus the domain of each function must be the same as the range of the other.

- Changing functions and variables in the model may be constrained to evolve in certain ways: i.e. they may be constrained to vary continuously or to have particular allowed values.

- Since the only driving mechanism available is the function collectively known as *the laws of physics* (as they exist rather than as we understand them), $\mathbf{L_P}$, the constraint that $\mathbf{L_E} = \mathbf{L_G} = \mathbf{L_P}$ may be imposed.

  In Section 5, we will consider some potential specific constraints of the laws of physics.

4.2 Virtual interaction

The second class of interaction to be considered is *virtual interaction*, which may be mediated by *symbolic representation* and *communication* (thus allowing the domains/ranges to be mismatched). Here, we define a *symbol* using Turing's 1936 definition: a letter or sign taken from a finite alphabet to allow distinguishability. We define communication using Shannon's communication theory: the sending of a *message* to a *receiver* via a (potentially) *noisy channel* (Shannon and Weaver, 1949). See Figure 2.

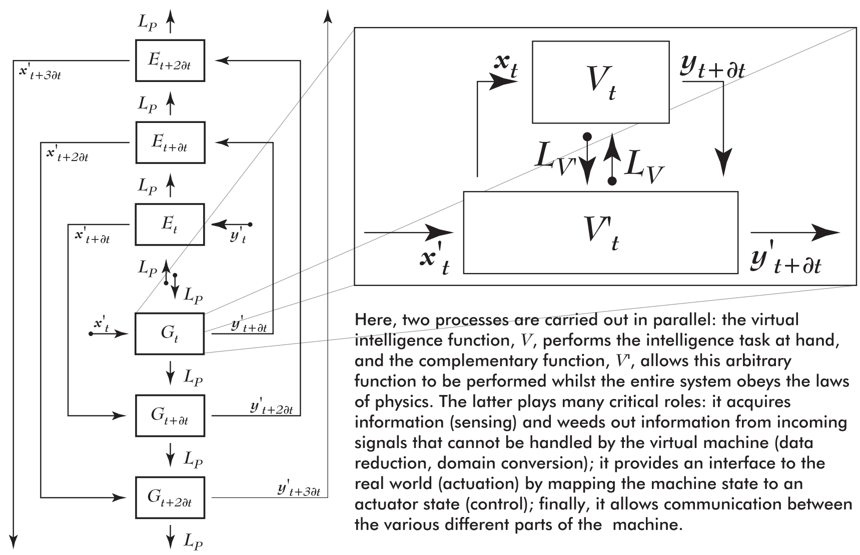

Figure 2: In deterministic virtual interaction, real interaction is also taking place. However, the output from the environment is not the same as what is being input to the function that we deem intelligent: the virtual machine with function $\mathbf{V}$. For this to be the case, we model the interaction as including a complementary function ($\mathbf{V}'$), that filters the input from the outside world, feeds it into the virtual machine, processes the remainder, and combines the virtual machine result with its own to produce the final output. The virtual machine and its function are normally analyzed by computer scientists, whereas the complementary functions are the province of electrical, electronic, mechanical, and other engineers.

If we define $\mathbf{V}$ as the *virtual intelligence function*, analagous to $\mathbf{G}$ but for the virtual case, some major constraints are lifted, with important consequences. First, x is no longer constrained to be equal to x' (likewise for y and y'). What this means is that the inputs and outputs may not be considered in their totality, but selectively (entire modalities may simply be excluded by x and y, or specific ranges within specific modalities may be excluded). This can be considered in two ways. From a design perspective, it means that behaviours (sensor-actuator pairs) may be considered identical even if the global inputs and outputs (the inputs and outputs taking all possible modalities into account) are very different.

For example, when the letter A is typed into an laptop computer, how hard the key is pressed (within a range) is irrelevant. Soft A and hard A are considered, within the virtual environment, to be identical inputs. Likewise, the brightness of the screen is irrelevant to the meaning of the letter A when it appears before the user: whether or not our laptop is in power-saving mode does not affect our perception of the output here. So we have virtual sensor-actuator pairs (letter is typed, appears on screen) that can be very different physically but are considered to be the same behaviour virtually. In this scenario, there are fewer sensor modalities available to the embodied intelligence than there are actual modalities of physical influence coming from the environment.

As a result of this, the actual output from the environment may be very different from the input received by the virtual embodied intelligence. They may be very different because they are not allowed to affect the working of the virtual machine at all (just as how hard a key is pressed is information that is not available to the computing machine within the laptop). Or they may be represented very differently. For instance, temperature, which may be continuously varying in the environment, may be represented as a number with just one decimal place in the machine. Mathematically, in either of these cases, the domain and range of the two functions may be different.

In order for all of the above to be true, a new function must be defined: the *complementary function* $\mathbf{V}'(x'_t - x_t)$ that ensures that, together with the intelligence function, the entire system obeys the laws of physics. The existence of this function can be considered to be a test of whether a system is virtual or not.

Another important constraint that is lifted is that $\mathbf{E}$ and $\mathbf{V}$ need not have any kind of conserved relationship, and $\mathbf{L_V}$ need not be the same as $\mathbf{L_E}$. Because only range/modality subsets of x' and y' attach to $\mathbf{V}$ and $\mathbf{L_V}$, the intelligence and adaptation functions are partially de-coupled from the environment. With the right machine and choice of sensor modalities, arbitrary choice of $\mathbf{V}$ and $\mathbf{L_V}$ may be made.

It is important to note that this arbitrary choice *depends* on the selection of inputs/outputs, since a real interaction (with adaptation function $\mathbf{L_P}$) *must* be taking place at the same time as this virtual interaction. Thus, it is only because of $\mathbf{V}'$ and $\mathbf{L_{V'}}$ that such freedom is allowed for $\mathbf{V}$ and $\mathbf{L_V}$.

4.3 Non-deterministic interaction

The argument so far assumes (as is generally assumed in all branches of physics, except quantum mechanics, which will be discussed in more detail later) that the physical evolution taking place is both a causal and a deterministic process. In this context, causal means that the state of the two systems $\mathbf{E}$ and $\mathbf{G}$ at a given time t is the direct and only cause of its state at time $t + \delta t$. In other words:

$$(\mathbf{E}_t, \mathbf{G}_t) \to (\mathbf{E}_{t+\delta t}, \mathbf{G}_{t+\delta t})$$

Deterministic means that the state at time $t + \delta t$ can be predicted from that at t. Non-deterministic means that it cannot. Note that there is an ambiguity inherent in this definition. It is not stated by whom or by what this prediction can be made, nor in what circumstances. It may be unpredictable in principle because the process has some kind of random element, one not following any principle or order, embedded in it. Or it may be non-deterministic in practice: because insufficient information is available to make a reliable prediction.

It may seem that an interaction may not be both causal and non-deterministic. However, from the perspective of conventional quantum mechanics, the two are compatible in the following sense: physics (in the form of the wave equation) causes a particular set of outcomes to be possible, but which of these outcomes actually takes place is not determinable in advance. This is the definition of a stochastic process. The word random, left undefined in the last paragraph, can be more clearly understood in this context. If the process of "choosing" between different possible outcomes is not determined by physics, then it must be determined by something outside or above physics: meta-physics.

Two examples of *in-practice* non-determinism may make clearer the distinctions between this and the *in-principle* variety described above. First, a chaotic system might be *practically* non-deterministic to an observer if it were impossible to get infinitely precise information about it: however, there may be no principle that says that such information could not be made available. Second, a quantum-mechanical system might be considered unpredictable by scientists because the required properties and variables to make the prediction cannot be measured without changing them.

As the physical model presented in this thesis is not based on the manipulation of information or prediction based on a model, the *in-practice* scenario does not qualify as non-deterministic. Consequently, the model described in this section only applies if the *in-principle* variety of non-determinism is true. This distinction becomes important when the issue of how the mathematical model relates to real physics is considered.

4.3.1 Model of stochastic interactions

Mathematically, a causal, deterministic process can be represented by a one-to-one mapping from the input to the output: for a given set of conditions, only one outcome is possible. Both the real and virtual interactions described in the previous sections are based on this type of mapping. A stochastic or non-deterministic interaction, on the other hand, must be represented as a one-to-may mapping. In this case the output (x' or y) cannot be represented by a single value. Instead it must be taken from a set. In principle, this set may have any cardinality (with the possible exception of being empty): it may be finite, denumerable or non-denumerable. In Figure 3, for simplicity, the set of possibilities is kept to just two.

As can be seen in the figure, there is not (by definition) a pre-determined timeline. Specifically, knowing the state of one system at one time no longer uniquely implies the state of the other at that time, nor its own state any time later. This can be understood by following the various allowed evolutionary pathways for the two systems. As time elapses, the number of states that each system may have evolved into increases. This may be considered a type of divergence, in that it directly prevents tight coupling between the two systems.

To allow the mathematical model to take this type of evolution into account, a new *random variable*, z, can be defined that chooses which of the possible physical outcomes occurs: this variable is entirely independent of x, y, x', and y'.

Such a scenario may be considered to be a type of *double* virtual interaction. Functions **G** and **E** are constrained (by definition) to only act on input from each other (x, y, x', and y'). And yet, somehow, there is a function in existence that operates on z. As our model

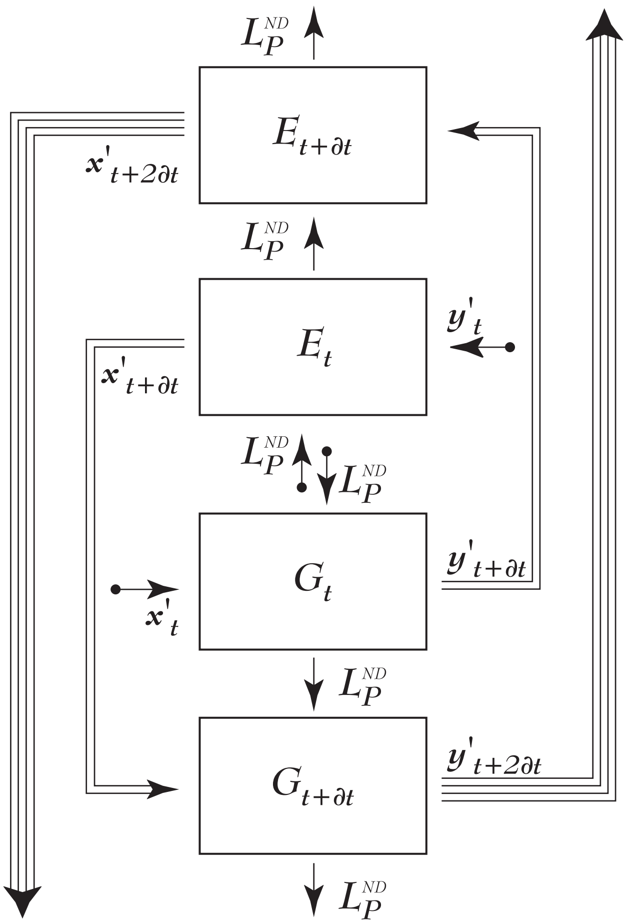

Figure 3: In a non-deterministic universe, more than one outcome may be possible for a given physical event (the non-deterministic laws of physics are used here: L_P^{ND}). Two are shown here for simplicity. Thus, the coupling relationship between the two systems may change over time depending on which course (path) events take. Since the choosing of the path is metaphysical (not produced by object or environment), such an interaction cannot be considered real as it has been defined here.

is defined, the only way this is possible is if $\mathbf{E}$ and $\mathbf{G}$ are mapped to virtual functions $\mathbf{V^E}$ and $\mathbf{V^G}$ and their complementary functions, $\mathbf{V'^E}$ and $\mathbf{V'^G}$, are considered to keep track of z (see Figure 4). Here, $\mathbf{V^G}(x) \rightarrow \mathcal{Y}$, and $\mathbf{V'^G}(\mathcal{Y}, z) \rightarrow y'$. Crucially, z is metaphysical here: it's value is determined by some process that is not entirely dependent on, or related to, any of the physical laws or variables.

Given this analysis, only deterministic physical processes can be considered to fall into the class of *real interactions* as defined above.

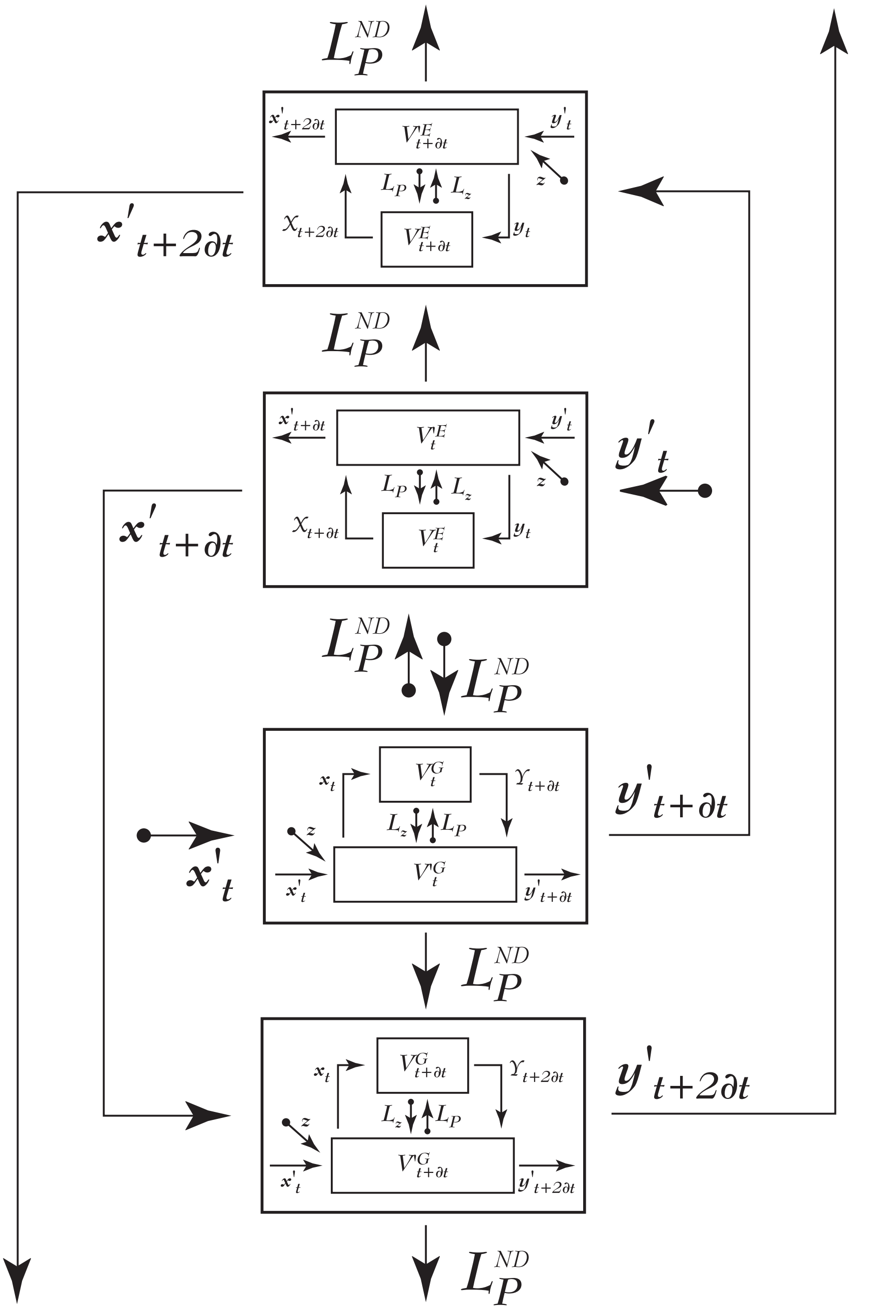

Figure 4: Instead, the non-deterministic case can be considered to be a kind of double virtual interaction, where the virtual machine is that obeying the deterministic physical laws that produce a set of possible outcomes, and the complementary function chooses from this set based on some metaphysical random variable, z.

4.4 Real versus virtual interaction

To summarize this section, it is worth going back to our laptop example and comparing the real and virtual interaction analyses. In the former case, the machine may be considered just like any object. It is something that moves under gravity, has resistance, gives off heat, makes noise when touched etc.. It can be used as a paper-weight or a doorstop. All of these uses of the machine are covered. In the virtual analysis, only the tapping of the keys and the pixels on the screen are considered to be part of the machine's intelligence function: $\mathbf{V}$. All other aspects of its behaviour, which combine to give it the global function $\mathbf{G}$, are included in $\mathbf{V}'$.

Thus, for simulated robots, artificial life, and other applications where the interaction is purely symbolic and has no real meaning (apart from for display purposes) outside the machine, $\mathbf{V}$ is all that's important: the complementary function is irrelevant (as it is normally treated in computer science). For real robots in a real world, however, $\mathbf{G}$ is critical. Thus, $\mathbf{V}'$ must be taken into account when designing and building them.

5 The interface with physics

As described so far, the model is entirely general, in that we have not assigned the various functions or variables as belonging to any specific sets. It is only with this assignment that the representational and computational power of the model can be determined and (as required) compared to other computational models (such as the Turing machine). In order to provide the bridge between the theoretical and the physical, these assignments must come from our understanding of the physical laws. The relevant correspondences are outlined below.

The first important physical question that must be answered in order to allow comparison between various models has already been alluded to in Section 4: this is the issue of whether physics is entirely deterministic, or whether it is not. As discussed earlier, this information allows us to know whether real interaction actually exists, and therefore which model should be used.

Second, x, x', y, y', are place holders for the multidimensional space that includes all the physical state variables, and t is the placeholder for time. From a mathematical perspective we can ask two very simple questions about these variables that allow us to begin to understand their representational power. Are their sets finite or infinite? If infinite, do they have cardinality $\aleph_0$ or $\aleph_1$? These questions are crucial, because the answer determines whether or not the physical computation is less powerful, more powerful, or as powerful as (for instance) the Turing machine.

For instance, if it turns out that $\mathbf{G}$ can map a continuous variable onto a continuous variable ($|\mathcal{X}| = |\mathcal{Y}| = \aleph_1$) then this is the equivalent of a machine with an infinite symbol set and an infinite rule book: this is, specifically, not allowed with a Turing machine and represents the super-Turing case (it potentially contains all the mappings available to the Turing machine as well as others). For $|\mathcal{X}| = |\mathcal{Y}| = \aleph_0$, and $|\mathcal{X}| = |\mathcal{Y}| = N_{MAX}$ (a finite set), the machine would be Turing-equivalent, or sub-Turing, respectively.

With some rearrangement, the questions about cardinality can be turned into physical ones and then fed back into the model. First, are the multi-dimensional physical state variables and time—or, more simply, is space-time—continuous, discrete (i.e. there is a minimum δx or δt), or arbitrarily discrete (i.e. there is no specific minimum δx or δt but the variable is still not defined as continuous). Second, are the physical systems in question bounded or unbounded? For this latter question, in the case of the embodied intelligence, the answer is given in the definition: it *is* bounded. In the case of the environment, however, the question is still not decided among physicists.

We will leave this discussion here, as a full explanation of the relationships between the various physical and mathematical options is beyond the scope of this paper. However, we hope that it is, at least, clear that an interface between the theoretical and physical paradigms does exist here.

6 Discussion

In the brief discussion of comparisons above, a practical matter is not made explicit. The question we are asking is whether or not $\mathbf{G}$ may be replaced with a Turing machine. In fact, as discussed in Section 2, we know it cannot be, because there is no mechanism for getting information in and out of the machine from the real world. Thus, it must be the complementary function, $\mathbf{V}'$, that allows us to feed our Turing machines (or digital computers) with the symbols they need, and to use the resulting symbols to produce an output. Thus, we can explicitly say that any symbolic interaction is, by definition, a virtual interaction, and the complementary function is crucial to its success. What, in practice, is $\mathbf{V}'$? It is the machine's analogue sensors and actuators and enablers: mechanics, optics, hydraulics, heat-sinks, etc..

The same thing might also be said of our bodies, which not only provides our conscious mind with sensors and actuators, but also the un/subconscious mind which is not normally considered in AI. Other bodily systems—the limbic system, the spinal chord, even basic organs like the heart and lungs—all contribute to our behaviour to a greater or lesser extent. Using the virtual interaction model, we can see that what many people consider to be artificial intelligence, particularly purely software-based approaches, concerns only $\mathbf{V}$ and not $\mathbf{V}'$.

What is interesting here is not only that this $\mathbf{V}'$ exists, but that it may be as important as $\mathbf{V}$ for some applications.

This fact is increasingly being recognised by roboticists. For instance Williamson, who worked on Rodney Brooks "Cog" project, was charged with engineering the robot's limbs in such a way as to ease both the information-processing and energy burden they represented for the machine as a whole. A mechanical engineer by training, he designed Cog's arms and wrists so that they were compliant (Williamson, 1995) and could respond *mechanically* to changes in the environment rather than purely through conventional sensor-processor-actuator loops. The result was not only a more mechanically-efficient and natural-looking movement, but considerably lower computational overheads. Others interested in exploiting a balance between information processing and physical (in this case, mechanical) computation include Lungarella and Berthouze (2002), who have looked at how temporarily restricting the degrees of freedom of a mechanical system can improve a robot's ability to learn how to manipulate it, and Pfeiffer who gives numerous examples of the importance of mechanical design to machine intelligence (Pfeifer, 2002).

This balance may also be said to be the concern of the neuromorphic engineering field pioneered by Mead. This work often goes much further, blurring the interaction boundaries. Projects like Harrison and Koch's all-analogue fly vision system and robot controller (Harrison and Koch, 2000), Hasslacher and Tilden's analogue walking robots based on the nervous systems of small animals (Hasslacher and Tilden, 1995), or Lewis's bipedal robots based on central pattern generators (Lewis *et al.*, 2001), all have in common that there cannot be said to be a clear line between sensor, processor, and actuator. Indeed, there cannot be said to be a clear line between software and hardware. Whether their systems can also be considered to perform real interaction or not is complex (and will not be considered here), but their machines certainly seem to address the complementary function more fully.

Finally, one could also argue that full understanding of real interaction, virtual interaction, and the complementary function might go some way to addressing Brooks' recent question about the relationship between matter and life (Brooks, 2001). In essence, we may just need to go further to try to understand a whole brain and body (both $\mathbf{V}$ and $\mathbf{V}'$) in order to understand how a creature works, rather than considering mind alone.

7 Current and future work

From the description above, several projects suggest themselves: some of which are currently underway and others that might usefully be done in future. The first of these, currently in progress, is the analysis of the model under constraints as suggested by current (and often contradictory) interpretations of the physical laws, and comparisons with the Turing model given the options that arise from these. In particular, issues related to quantum mechanics are problematic. Whether the universe may be considered discrete or continuous is not a trivial problem: there is no consensus here. Likewise, there are still many who believe that the current consensus in quantum-mechanics—of which it may be said that the theory is sound, but the philosophical underpinnings not—will eventually have to change. We are currently developing a map of these ideas so that, as the physics develops, its impact on the question of intelligence can be seen clearly. This work is currently being written up.

We are also trying to identify those applications of intelligence where a mismatch in representational power may be important. Clearly, information-theoretic approaches are appropriate in many engineering scenarios: today's engineering is based on such approaches. We are currently drawing up a simple specification, in terms of the types of functions that may be important and the type of stimuli that may need to be represented, that should allow engineers to understand where a more physical approach may be warranted. This work is also at the draft stage.

As a longer term goal, we would like to be able to help the engineer who, using the test outlined about, has determined that the conventional design approach is inappropriate for the task in hand. This involves the development of a set of design rules that would determine how analogue to digital conversion layers are placed in a given system: i.e. how to correctly balance the analogue and digital processes in order to maximise efficiency for a given task. Note: in some cases it might be expected that *no* conversion is the best option, thus suggesting an analogue-only system.

Acknowledgements

The author would like to acknowledge: her supervisor at the Open University, Jeffrey Johnson; the help of the reviewers; Christof Koch of the California Institute of Technology for important discussions and reading the manuscript; her colleagues at Imperial College; and numerous other scientists and engineers who have contributed to this work through inspiration, discussion, and debate.

References

Blum, L., Shub, M., and Smale, S. (1989). On a theory of computation over the real numbers; NP completeness, recursive functions and universal machines. *Bulletin of the American Mathematical Society*, **21**(1), 1–46.

Brooks, R. (2001). The relationship between matter and life. *Nature*, **409**, 409–411.

Chua, L. (1998). *CNN: A Paradigm for Complexity*. World Scientific.

Dreyfus, H. (1972). *What computers still can't do: A critique of artificial reason*. MIT Press.

Farhat, N. (1997). Cognitive networks for ATR: The roles of synchronicity, bifurcation and chaos. Technical report, Office of Naval Research. Final Report.

Farhat, N. and Wen, Z. (1995). Large-scale photonic neural networks with biology-like processing elements: the role of electron trapping materials. *Proceedings of the SPIE*, **2565**.

Gödel, K. (1931). *On formally undecidable propositions of Principia Mathematica and related system*. Dover.

Harrison, R. and Koch, C. (2000). A silicon implementation of the fly's optomotor control system. *Neural Computation*, **12**(10), 2291–2304.

Hasslacher, B. and Tilden, M. (1995). Living machines. *Robotics and Autonomous Systems*, **15**, 143–169.

Laforte, G., Hayes, P., and Ford, K. (1998). Why Gödel's theorem cannot refute computationalism. *Artificial Intelligence*, **104**(1-2), 265–286.

Lewis, A., Hartmann, M., and Etienne-Cummings, R.and Cohen, A. (2001). Control of a robot leg with an adaptive aVLSI CPG chip. *Neurocomputing*, **38-40**(1-4), 1409–1421.

Lungarella, M. and Berthouze, L. (2002). Adaptivity through physical immaturity. Oral presentation at 2nd International Workshop on Epigenetic Robotics.

Maass, W. (1997). Analog neural nets with gaussian or other common noise distributions cannot recognize arbitrary regular languages. In *Electronic Colloquium on Computational Complexity Report*. TR97-052.

Mead, C. (1989). *Analog VLSI and Neural Systems*. Addison Wesley.

Penrose, R. (1989). *The Emperor's New Mind*. Oxford University Press.

Penrose, R. (1997). *The Large, the Small, and the Human Mind*. Cambridge University Press.

Pfeifer, R. (2002). On the role of embodiment in the emergence of cognition: Grey Walter's turtles and beyond. Invited Speaker presented at Biologically-Inspired Robotics: The Legacy of W. Grey Walter.

Shannon, C. and Weaver, W. (1949). *The mathematical theory of communication*. University of Illinois Press.

Siegelmann, H. (1999). *Neural Networks and Analogue Computation: Beyond the Turing Limit*. Birkhäuser, Boston, MA.

Tegmark, M. (2000). The importance of quantum decoherence in brain processes. *Physical Review*, **61**, 4195–4206.

Turing, A. (1936). On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, **42**, 230–265.

Williamson, M. (1995). Series elastic actuators. Technical Report 1524, MIT AI Lab. AI Memo.