Being Analog

Sunny Bains
Electrical and Electronic Engineering Department,
Imperial College London, United Kingdom,
s.bains@imperial.ac.uk

February 22, 2011

Abstract

The conventional wisdom has been that artificial intelligence is all about algorithms, and therefore restricted by the Turing model of computation. In fact embodied machines (those that interact with the outside world via sensors and actuators) cannot reasonably be modelled as being 'digital' in the conventional sense, or even as Turing machines. Far from being a disadvantage, by understanding and exploiting analog nature of these machines we have the opportunity to increase power efficiency, improve learning and discrimination, and provide better adaptability to changing circumstances.

1 Big Artificial Intelligence

Since long before Roger Penrose wrote his misleading book, *The Emperor's New Mind* [6], people have been confused about what they mean by artificial intelligence (AI). Sometimes they mean entirely *virtual* intelligence: that is, intelligence that only operates inside the closed-off virtual world of a machine (like artificial life). Sometime they mean intelligent systems that take information from the outside world and pass commands out to it, but entirely mediated by fixed analog-to-digital (a/d) converters, feature extractors and so forth that decide what input the 'intelligence' should operate on before it gets to see it.

But when I use the term AI, I mean what probably we all think of as the eventual goal of artificial intelligence: a thinking, acting, sentient android or robot that can adapt to new tasks and learn—as we do—through experience. To differentiate, I use the term Big AI. Of course, it would be pointless if such machines were identical to us. They would be faster or stronger or cleverer than we are. However, they would use many of the same mechanisms for intelligent behavior as we do, because these mechanisms have been tested by evolution and found worthy.

I would argue that interest in Big AI requires the abandonment of the traditional practice of engineering, while in another sense it takes the discipline to a new level. Traditional engineering involves a specification, which might be something like: "build me a machine that can do *this task*, and that satisfies the following constraints." The constraints could be power efficiency, size, safety, compatibility with other systems, whatever. The task in question could be a single task or it could be a *finite* set of tasks, but whatever it is, it is specific. Thus, without such a specification, we could not set about building a system in the old way.

So how would we specify a human, or its machine equivalent, an android? It's true, what a human can do is constrained. But is it finite? Could we write down every little task we might want a machine to be able to perform and then program it to do that in every conceivable set of circumstances? Of course not: we wouldn't try.

This is why machine learning—both for the most sophisticated and trivial of applications—is such a huge topic of interest. We understand that machines are going to have to learn what to do as well as how to do it. The building of systems that can't do what's necessary, but which can learn, is creating a new level of engineering: a kind of meta-engineering if you like.

2 The Analog Shell

But this shifting of the problem from *knowing* what you need to *learning* it does not just apply to the intelligence side, *it applies to the sensors and actuators too.* If you don't yet know all of the environments a machine will have to work in during its lifetime, or all the tasks it will have to perform, how can you accurately decide how much you need from its sensors? And how will you decide which systems in the brain or nervous system should receive the sensor data and which do not need it? And how will you even decide what constitutes a sensor and what does not?

The fact is that physical objects 'process' physical information whether they want to or not. They are inherently coupled to all of the physical forces around them. Windows, for instance—by which I mean the glass things that let you look out of buildings, not the operating system—vibrate in the presence of noise. This is why you can shine a laser onto them and, based on the changing reflection, use them as a kind of microphone. Windows are not sensors by the term's conventional meaning, but they *do sense.* Neither are they actuators in the conventional sense, but they *do actuate.*

This is not just an abstract argument. Parts of the body that would not be accepted as 'sensors' by any conventional engineer are, in fact, used in human cognition. The most dramatic demonstration of this that I've come across is the percussionist Evelyn Glennie who, though born hearing, became profoundly deaf. According to her own website [4], "Evelyn spent a lot of time when she was young (with the help of Ron Forbes her percussion teacher at school) refining her ability to detect vibrations. She would stand with her hands against the classroom wall while Ron played notes on the timpani (timpani produce a lot of vibrations). Eventually Evelyn managed to distinguish the rough pitch of notes by associating where on her body she felt the sound with the sense of perfect pitch she had before losing her hearing. The low sounds she feels mainly in her legs and feet and high sounds might be particular places on her face, neck and chest."

So our brains are able to leverage incidental information (information that most engineers would describe as noise) and use it to solve problems. Why would we want to prevent the machines we create from being able to do this as well? Because as soon as you start to over-engineer—that is, be too specific about what you want a particular part of your machine to achieve and how you expect it to perform—you risk preventing that leg or ear or muscle from being used to help perform other tasks that you haven't thought of yet.

From a practical point of view, what I am arguing is that if you have a nice analog sensor (as most are), why are you ignoring part of what it's trying to communicate by forcing it to 'speak' through the filter an analog-to-digital converter? Not only does this force the signal into permitted digital states, but it can mask the underlying signal shape and important temporal dynamics.

Neuromorphic engineering—the goal of which is to create mainly (though not exclusively) analog circuits and systems that behave with similar dynamics to those found in the brains, senses, and nervous systems of real animals—is full of examples where processing the analog signal *in the analog domain* can give you a huge advantages in power and speed. For a full description of how these advantages are possible, how the approach relates to Big AI, many examples of successful neuromorphic engineering projects, and a description of some of the alternative computing technologies that can be thought of as being compatible with the approach (including optical) please see [2] and Chapter 2 of [3] and references therein.

A colleague, Chris Toumazou, describes this problem of getting information from the analog real world to the virtual world of computers as *the analog shell.* What's interesting about this metaphor is that it implies that you need to think about where you put barrier between the analog and digital domains: too early and you either throw away information or leave yourself with lots of (expensive) digital signal processing to do.

However, I prefer to think of it not a shell, but maybe a set of shells (like Russian dolls). At each level of processing, the information is more processed, more abstract, less analog, more digital, until eventually you can safely do a straight a/d conversion (or even just treat the analog information as digital) without losing anything.

The advantage of this is that you don't have to approach the problem of Big AI in either an entirely analog or digital way: you can be pragmatic about it. It may make sense for those working on high-level reasoning or mathematics to work using more conventional computing (digital/Turing) models. Likewise, however, you would not want to have to use entirely digital technology for the huge mass of signals being received by an android body: analog solutions will probably be more appropriate for the initial signal processing. The difficult engineering problem can then be defined as making sure that the information moves smoothly and appropriately from one regime to the other.

3 Physical beings, physical computation

However, I would argue that analog computation is just a subset of the bigger challenge to traditional approaches to AI. Certainly, the misunderstanding of continuous approaches is a major problem for many in this field: hence the name of the paper. But the deeper problem is that computer scientists forget that not every system has to turn a task into manipulation of 'information' in order to perform useful processing.

There are numerous examples of what I would call 'physical computing': using the physics of a device to straightforwardly process a signal. Familiar examples would include mechanical scales, sundials, even lenses that can take an image and Fourier Transform it *at the speed of light.* Such processes are inherently efficient. Which is the quickest way to make a robot limb safer: by calculating the forces impinging on it, working out from the context what is going on, and giving a command to withdraw the limb; or by putting springs in the limb, so making it compliant? What is the fastest way to find matching pictorial elements on images, by cutting them into strings of bits and using complex algorithms to reconstruct their geometry and make nearest-neighbor comparisons? Or to optically filter the Fourier Transform of one image with another in the time it takes to refresh a display (optical correlation)?

It is true that physical solutions are rarely as widely applicable as their information-based counterparts, just as custom chips are usually more efficient but less adaptable than general-purpose processors. But with a problem as difficult as trying to fit the intelligence, acuity, and dexterity of a person into a machine, surely it is necessary to take advantage of every efficiency available?

And, even if this were not the case, even if we saw no advantage whatsoever in using efficient analog, optical, mechanical or chemical processing, even then, there is a problem. Because androids and robots do *not* live in a virtual world and therefore *cannot* be understood purely as Turing machines. *By definition,* any robot that acts in the world is acting in an analog way (if analog has any meaning at all, those of you want to argue about discrete space-time models can see the arguments in my thesis [3]). Wheels spin in through an uncountably infinite number of states, limbs move through an uncountably infinite number of points.

For computer scientists, this can be a terrifying idea. Essentially, what I am claiming is that people are not computers. I'm not the first, of course: Turing said it himself and there is a whole field of people who are interested in the idea of applying dynamical systems theories to both 'real' and artificial intelligence.

But again, this should not be seen as a problem for AI researchers, but rather an opportunity. Embodied artificial intelligences *need not be constrained* by the computational limits of Turing machines. Researcher Hava Siegelmann has shown that recurrent analog neural networks (which seem to be the kind of networks that our brains use) can perform functions that would normally be considered super-Turing [7].

4 The drawbacks

Where all the evidence points to the idea that physical computation can improve computational power, energy efficiency and speed, why has the switch to analog and physical approaches been, relatively speaking, so slow? There are many reasons. It is partly because the theoretical framework isn't there yet. A goal of my research [1] was to start to build a bridge between the Turing and physical worlds, but it will take concentrated practical and theoretical work to develop a set of robust guidelines that help engineers to understand which problems are best tackled in an analog way, in a digital way, or in a more direct physical way.

Another problem is that, to be blunt, we've all got a bit lazy. Doing research in analog or dynamical systems is complex (which I mean in both the everyday and mathematical senses of the word) and mathematically challenging. Programming is much easier (and doesn't require you to send things to be fabricated). There is an increasing shortage of analog electronics engineers, which means that—even when digital solutions are less efficient, more expensive, etc.—they are sometimes easier to implement.

Finally, a major problem—at least in my experience—has come from ignorance within the computer-science community (and electrical engineering too). People think that the assumptions and approximations made in the Turing machine model and communications theory can be applied to all of physics. This is simply not true.

In a way, this is the opposite of the mistake that Penrose makes: he assumes that computers are not physical, and therefore entirely constrained by Turing. What I am saying is that this constraint on computers may be true, but only applies to machines that operate in an entirely virtual world.

Yet another part of the problem may be the fact that we all understand the advantages of digital for communication and media. I'm not arguing that digital isn't (generally) a better technology to use in these fields. But the problem of trying to communicate and receive a specific message is completely different than the problem of an intelligent agent has when trying to suck as much information out of the environment as its sensors will allow in order to perform an *as yet unknown task.*

To say it another way, communication theory and the Turing model both have, as inherent assumptions, that there is a *known* acceptable level of approximation that can be tolerated. This assumption does not hold in Big AI. For some task a single bit response from a sensor may be sufficient. For other tasks many bits may be needed, with sophisticated averaging over time and space. The intelligent system must learn what it needs for what tasks, and what information it can afford to throw away. If it doesn't, it either risks not having the information it needs when necessary, or wasting a lot of time and energy processing signals at resolutions that are simply too high.

5 Moving forward

There are many avenues that can progress the physical computation agenda in AI, and people are working on a number of them. As well as the other disciplines I've described (neuromorphic, optical, mechanical) there are relatively new approaches to computing, like probabilistic electronics that (see e.g. [5]) aim to adjust the power devoted to a task to the need for precision in that task.

From my point of view, however, that harmonizing framework, the pulling together of one discipline from many disparate fields, that is missing. Perhaps it is too early for that.

While we are waiting, however, the best thing that we can all do, as computer scientists, as roboticists, as electronic engineers, is to keep an open mind. We should be constantly questioning whether the theories, methods, and approaches we are used to applying are really suitable for the job in hand. Do the assumptions that the theories were based on hold for this task? We should keep looking to biology for alternative solutions. How does the brain/spinal cord/retina solve this problem? And we should keep asking ourselves whether we are trying to be too controlling as engineers. Are we hardwiring behaviors rather than making it possible for the machine to learn to do what we need it to?

The technical challenges are not insubstantial. But before we can start to tackle them, we must open our eyes to the fact that Big AI really is not the same as the other engineering fields we are used to.

Further Reading

For brevity, references have been kept to a minimum in this paper. For further discussion of the practical problems of, and technologies that can be used in, Big AI, please see [2] and Chapter 2 of [3]. For further discussion of the theoretical issues surrounding Big AI, see Chapter 3 of [3]. Subsequent thesis chapters describe a simple model that incorporates the Turing machine into physical computation, and discuss how interpretations of physics may affect the theoretical implications of this model. The thesis and all papers are freely available at: `http://www.sunnybains.com`.

References

[1] Bains, S. (2003). Intelligence as Physical Computation, AISB Journal (1), pp. 225–240.

[2] Bains, S. (2004). Extending neuromorphic engineering beyond electronics, Proc. Brain Inspired Cognitive Systems, Stirling.

[3] Bains, S. (2005). Physical computation and embodied artificial intelligence, PhD Thesis, The Open University.

[4] Glennie, E. (1996). Evelyn's Hearing, at `http://www.evelyn.co.uk/hearing.htm`, accessed 2004.

[5] Palem, K.V. 2005. Energy aware computing through probabilistic switching: a study of limits, IEEE Trans. Computing, pp. 1123 – 1137.

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

[7] Siegelmann, H. T. (1999). Neural Networks and Analog Computation: Beyond the Turing Limit, Birkhäuser, Boston.