It Is Time To Investigate Bio-Inspired Computing Systems

Hugues Berry Olivier Temam
INRIA Saclay, France, {*first_name.last_name*}@inria.fr

The dissipation bottleneck, which slowed the progress of clock frequency and shifted computing systems towards multi-cores, was a reminder that the smooth evolution of technology we have enjoyed for decades may not last forever. Therefore, investigating future and alternative technologies and how they can be leveraged for designing and programming computing systems should not be labeled as exotic (if not useless) research: it comes out of a very practical, if not industrial, concern to anticipate drastic changes soon enough to be ready when needs be. For instance, research on parallelizing compilers and parallel programming models has intensified only when multi-cores became mainstream, and it is not yet mature in spite of strong industry needs.

Whether future technologies will be ultra-small CMOS transistors, nanotubes, or even individual molecules or biological cells, these elementary components all share several common properties: they come in great numbers, they won't be much faster or may even be way slower than current transistors, they may be hard to precisely lay out and connect, and they may be faulty.

The key question, then, is how can one design and, more importantly, program a computing system using billions of such components while we are not even capable of harnessing a few hundreds traditional cores ?

Now, once one starts going down that path, it is almost irresistible to observe that nature has found, with the brain, a way to leverage billions of components with similar properties to successfully implement many complex information processing tasks. While suggesting to design computing systems which somehow imitate the brain is such an old cliché that most computer scientists are embarrassed to bring it up, biologists may be about to force us to reconsider. That biologists have made tremendous progresses in understanding how at least some parts of the brain works is not yet well-known to computer scientists. And it could be time to leverage some of these progresses for designing at least special-purpose computing systems.

One particular brain function, vision, is so well understood that biologists are now starting to write software models that rebuild them from the ground up using individual neurons [1]. The principles are quite different from artificial neural networks (ANNs), which essentially describe the behavior of some of these building blocks rather than how they can be architected together to implement complex functions. Moreover, the understanding of the implementation of vision processing opens a window on several general principles that seem to be in play for "designing" and "programming" information processing systems using billions of slow, faulty and irregularly connected components.

One of the main principles is automatically *abstracting* and then manipulating increasingly complex notions (e.g., pixels, then segments, then elementary shapes, then complex shapes) by hierarchically combining a small set of simple, and always *local*, operators (such as `max`, `sum`,...). Therefore, part of the "programming" is generic and lays in the architecture itself, and the rest in the training. Unlike in most ANNs, the training is unsupervised, complex notions emerge simply based on their frequency of occurrence, through permanent competition among the different building blocks. The same process also explains the ability of biological neural networks to specialize on a given task (e.g., certain types of images). Surprisingly, these simple principles seem sufficient to yield complex image recognition capabilities. Potentially, they can be extrapolated to, at least, any pattern matching task, and combinations of such tasks.

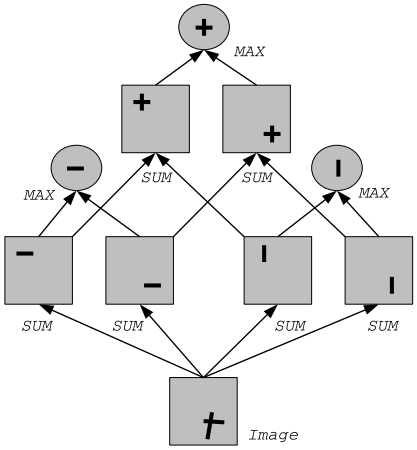

Figure 1. *Automatic abstraction through hierarchical networks.*

Figure 1 illustrates the approach. By connecting together small neural networks (using the `max` operator) which identify segments of different orientations or positions (using the `sum` operator), it is possible to create a network which detects a segment independently of its orientation/position (the same principle can yield size-invariant networks). Simultaneously, networks which detect two overlapping segments can detect a more complex shape like a cross. Combining networks detecting increasingly complex shapes with invariance networks can yield a network capable of replying to questions such as "is there a cross within the image", after training the global network simply by exposing it to a large set of images containing all kinds of segments (and not necessarily the target image, e.g., a cross).

Currently, biologists still use precisely laid out and reliable neurons to rebuild such complex functions in their models. Our goal is to show that randomly structured [2] and faulty neurons are compatible with this approach, and then to implement the same approach with other technologies.

[1] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):411–426, 2007.

[2] H. Berry and O. Temam. Modeling Self-Developping Biological Neural Networks. *Neurocomputing*, Vol. 70, Issue 16-18, 2007.