Computational Matter: Evolving Computational Solutions in Materials

Julian F. Miller
Department of Electronics
University of York, York, UK.
julian.miller@york.ac.uk

Hajo Broersma
Faculty of Electrical Engineering, Mathematics and Computer Science
University of Twente, NL
h.j.broersma@utwente.nl

ABSTRACT

Natural Evolution has been exploiting the physical properties of matter since life first appeared on earth. Evolution-in-materio (EIM) attempts to program matter so that computational problems can be solved. The beauty of this approach is that artificial evolution may be able to utilize unknown physical effects to solve computational problems. This methodology is currently being undertaken in a European research project called NASCENCE: Nanoscale Engineering for Novel Computation using Evolution [1]. In this project, a variety of solutions to computational problems have been evolved using mixtures of carbon nanotubes and polymers at room temperature and also with gold nanoparticles at temperatures less than one Kelvin.

Categories and Subject Descriptors

C.1.m [**PROCESSOR ARCHITECTURES**]: Hybrid systems; D.1.2 [**Software**]: Automatic Programming

Keywords

evolutionary algorithms; material computation; evolvable hardware

1. INTRODUCTION

Darwinian evolution can be viewed as an algorithm which exploits the physical properties of materials. One of the aims of the NASCENCE project is to assess the ability of EIM as a methodology for solving a wide variety of computational problems. One of the unique features of EIM is that it can exploit physical processes that a designer may either be unaware of, or not know how to utilize [5]. Exploiting materials may enhance the evolvability of evolutionary algorithms since subtle physical effects may allow beneficial transitions in the underlying fitness landscape. It may be possible to construct entirely novel physical computational devices using this approach.

GECCO '15 July 11-15, 2015, Madrid, Spain

ACM ISBN 978-1-4503-3488-4/15/07.
DOI: http://dx.doi.org/10.1145/2739482.2764939

2. CONCEPTUAL OVERVIEW

EIM is a hybrid system involving both a physical material and a digital computer. In the physical domain there is a material to which physical signals can be applied or measured. These signals are either input signals, output signals or configuration instructions. A computer controls the application of physical inputs applied to the material, the reading of physical signals from the material and the application to the material of other physical inputs known as physical configurations. A genotype of numerical data is held on the computer and is transformed into configuration instructions. The genotypes are subject to an evolutionary algorithm. Physical output signals are read from the material and converted to output data in the computer. A fitness value is obtained from the output data and supplied as a fitness of a genotype to the evolutionary algorithm [5]. The overall concept is shown in Fig. 1. The evolved physical configuration instructions act as the "program". Once such a program is found it may be possible to build a special circuit that can supply this configuration, thus obtaining a device that can operate the program in a standalone manner. Such a physical device may potentially be able to compute at a very fast data rate and with low power consumption. Also, complex forms of computation may be able to be programmed into tiny amounts of materials (i.e. computationally powerful nanoscale devices). Configuration instructions may be digital or analogue voltages, signals of various frequencies and amplitudes etc. So an evolutionary algorithm might manipulate genetic data that defines characteristics of such signals.

3. HARDWARE PLATFORMS

In the NASCENCE project a variety of hardware platforms have been built to allow computer controlled application of signals to the material and for the response of the material to be measured. Some of the hardware systems (Mecobo) allow the possibility to map input, output and configuration terminals, signal properties and output monitoring capabilities in arbitrary ways [4]. Another system is based on commercial data acquisition (DAQ) hardware and software. Some platforms allow digital amplitudes (0 or 3.5V). Others allow analogue voltages to be applied to electrodes interfacing with materials and also analogue samples of the output response from materials. In addition, some setups allow evolution to decide which electrodes to supply inputs to. This is accomplished using computer reconfigurable analogue switches, which act as programmable switches that can be put under evolutionary control.

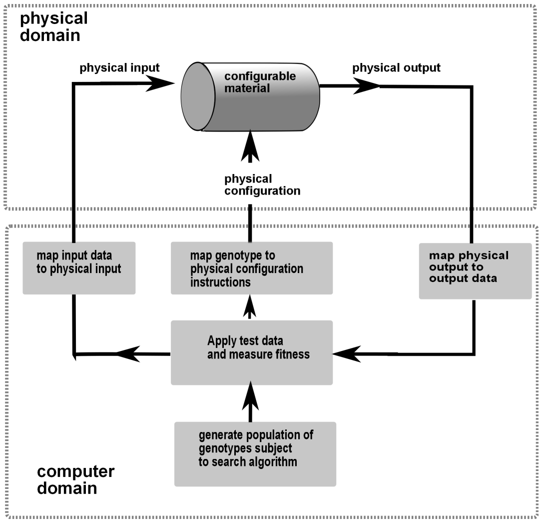

Figure 1: Concept of evolution-in-materio [5].

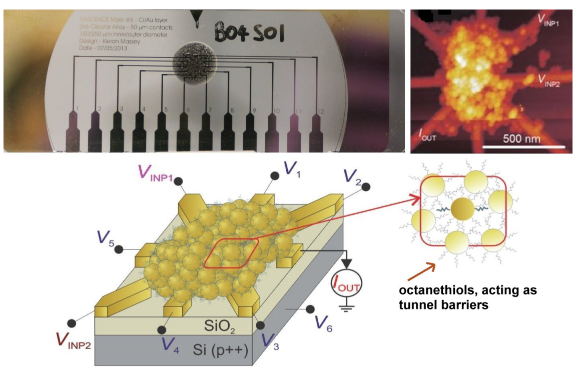

Figure 2: Electrode array with SWNT/Polymer sample (top left). Electrode array with Au nanoparticles randomly dispersed (top right). Schematic of nanoparticle array, showing input/output and configuration electrodes (bottom).

4. COMPUTATIONAL MATERIALS AND INVESTIGATED PROBLEMS

Two main electrode arrays with computational materials have been used in the project (see Figure 2):

- electrode array on a glass microscope slide with single-walled carbon nanotubes (SWNT) mixed with an insulating polymer (PMMA or PBMA)
- randomly dispersed gold nanoparticles (20nm) with 8 Ti/Au electrodes on a doped Si/SiO2 substrate.

A number of computational problems have been investigated with these devices. Some problems have no inputs and only require configuration signals and outputs (e.g. travelling salesman [2], bin-packing [7] and function optimization [8]). In the TSP, configuration voltages and where they connect to are evolved. A vector of voltages are read from the electrodes with as many elements as there are cities in the problem. Finally, the vector is sorted to obtain a permutation of cities. The length of the tour is used as the fitness. In bin-packing problems, in which hundreds of items have to be packed into bins, we used a single output from the electrode array and evolved many sets of configurations one after the other, to obtain the required number of output values (equal to the number of items to be packed). A mapping was used to determine from each output which bin an item would be packed in. Like bin-packing, function optimization merely requires as many outputs as domain vector elements. Large dimension problems require multiple chromosomes. Other problems attempted have inputs, e.g classification problems where the number of inputs equals the number of data attributes [9], Boolean logic circuits [3, 4], frequency classifiers which have one input to which various frequency square waves were applied [6]. In practice, one often has to devise complex input-output mapping functions to map genes and sampled output buffers to problem dependent variables.

5. CONCLUSIONS

Evolution-in-materio is hybrid of digital and analogue computing in which digital computers running evolutionary algorithms are used to configure materials to carry out analogue computation. This holds the promise of developing entirely new computational devices by directly exploiting physics in nanoscale systems and molecules.

6. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement number 317662.

7. REFERENCES

[1] H. Broersma, F. Gomez, J. F. Miller, M. Petty, and G. Tufte. NASCENCE Project: Nanoscale Engineering for Novel Computation Using Evolution. *International Journal of Unconventional Computing*, 8(4):313–317, 2012.

[2] K. D. Clegg, J. F. Miller, M. K. Massey, and M. C. Petty. Travelling salesman problem solved 'in materio' by evolved carbon nanotube device. In *Proc. Int. Conf. on Parallel Problem Solving from Nature*, volume 8672 of *LNCS*, pages 692–701. Springer, 2014.

[3] A. Kotsialos, M. K. Massey, F. Qaiser, D. A. Zeze, C. Pearson, and M. C. Petty. Logic gate and circuit training on randomly dispersed carbon nanotubes. *International Journal of Unconventional Computing*, 10:473–497, 2014.

[4] O. R. Lykkebø, S. Harding, G. Tufte, and J. F. Miller. Mecobo: A hardware and software platform for in materio evolution. In O. H. Ibarra, L. Kari, and S. Kopecki, editors, *Unconventional Computation and Natural Computation*, LNCS, pages 267–279. Springer International Publishing, 2014.

[5] J. F. Miller, S. L. Harding, and G. Tufte. Evolution-in-materio: evolving computation in materials. *Evolutionary Intelligence*, 7:49–67, 2014.

[6] M. Mohid, J. Miller, S. Harding, G. Tufte, O. Lykkebo, M. Massey, and M. Petty. Evolution-in-materio: A frequency classifier using materials. In *Proceedings of the 2014 IEEE International Conference on Evolvable Systems (ICES): From Biology to Hardware.*, pages 46–53. IEEE Press, 2014.

[7] M. Mohid, J. Miller, S. Harding, G. Tufte, O. Lykkebø, M. Massey, and M. Petty. Evolution-in-materio: Solving bin packing problems using materials. In *Proceedings of the 2014 IEEE International Conference on Evolvable Systems (ICES): From Biology to Hardware.*, pages 38–45. IEEE Press, 2014.

[8] M. Mohid, J. Miller, S. Harding, G. Tufte, O. Lykkebø, M. Massey, and M. Petty. Evolution-in-materio: Solving function optimization problems using materials. In *14th UK Workshop on Computational Intelligence (UKCI)*, pages 1–8. IEEE Press, 2014.

[9] M. Mohid, J. F. Miller, S. L. Harding, G. Tufte, O. R. Lykkebø, M. K. Massey, and M. C. Petty. Evolution-in-materio: Solving machine learning classification problems using materials. In *Proc. Int. Conf. on Parallel Problem Solving from Nature*, volume 8672 of *LNCS*, pages 721–730. Springer, 2014.