Heterotic computing examples with optics, bacteria, and chemicals

Susan Stepney[1], Samson Abramsky[2], Matthias Bechmann[3], Jerzy Gorecki[4], Viv Kendon[5], Thomas J. Naughton[6], Mario J. Perez-Jimenez[7], Francisco J. Romero-Campero[7], Angelika Sebald[3]

Departments of [1] Computer Science and [3] Chemistry, University of York, UK
[2] Department of Computer Science, University of Oxford, UK
[4] Institute of Physical Chemistry, Polish Academy of Sciences, Warsaw, Poland
[5] School of Physics and Astronomy, University of Leeds, UK
[6] Department of Computer Science, National University of Ireland Maynooth
[7] Research Group on Natural Computing, University of Seville, Spain

Abstract. Unconventional computers can perform embodied computation that can directly exploit the natural dynamics of the substrate. But such *in materio* devices are often limited, special purpose machines. To be practically useful, unconventional devices are usually be combined with classical computers or control systems. However, there is currently no established way to do this, or to combine different unconventional devices.

In this position paper we describe *heterotic unconventional computation*, an approach that focusses on *combinations* of unconventional devices. This will need a sound semantic framework defining how diverse unconventional computational devices can be combined in a way that respects the intrinsic computational power of each, whilst yielding a hybrid device that is capable of more than the sum of its parts. We also describe a suite of diverse physical implementations of heterotic unconventional computers, comprising computation performed by bacteria hosted in chemically built material, sensed and controlled optically and chemically.

1 Introduction

Unconventional computers promise great advantages, particularly by being able to perform embodied computation that can directly exploit the natural dynamics of the substrate [10,11]. But such *in materio* devices in practice are often limited, non-universal, special purpose. Additionally, they may struggle to perform some necessary functions that another substrate could handle with ease. For example, many devices suffer from the "wiring problem", that of moving information between parts of the computing device in a substrate with no natural means to implement any form of targetted long-distance communication.

To be practically useful, unconventional devices must usually be combined with other computers (possibly including classical computers) or control systems. It is thus advantageous to seek hybrids of separate devices that can each exploit their individual strengths to overcome any weaknesses in the other devices. For example, consider combining a complex substrate that can perform local computation, such as bacteria, with a different substrate that can readily implement communication "wires", such as optics. In addition to making certain forms of computation more efficient, combining non-universal computers can result in devices with more computational power than either alone.

The challenge is to develop a mature science of unconventional computation [13, 14], to complement that of classical computation. Progress has been made [12], but much remains to be done, in particular, in being able to combine disparate computational devices into a powerful whole. This latter point is the objective of heterotic unconventional computation.

The structure of rest of the paper is as follows. In §2 we overview some issues with single-paradigm unconventional computing, and illustrate these with some example paradigms. In §3 we outline our layered heterotic architecture, and describe how this architecture can address some of the issues of single-paradigm systems. In §4 we illustrate these claims with some example heterotic instantiations. In §5 we mention future steps needed.

2 Single-paradigm unconventional computers

In this section we consider the current state of single-paradigm unconventional computers, the issues they face, and provide some examples.

2.1 General issues

The issues with single-paradigm unconventional computers that are of most relevance to heterotic computation are:

Non-universality. Unconventional computers may not be universal, or may be universal only if used in an "unnatural" manner that obviates any unconventional advantage they exhibit for specific computations.

The "wiring problem". It is often difficult to move information between parts of the computing device, in a substrate with no natural means to implement any form of targetted long-distance communication. Yet there are alternative substrates (for example, optical) that excel at long range communication.

Information encoding. Binary logic is the default classical representation, but is not always the most appropriate for an unconventional device, which might more naturally support ternary or other multiple-valued encodings, or continuous variables. Many unconventional devices support only a unary (often analogue) encoding.

Hidden input/output computation. Inputs need to be prepared, and outputs decoded; this can often require considerable computation in its own right, and may be missed in traditional analyses of computational power.

Computation itself. Whether some candidate physical process can even be classed as computation, or is simply the system doing nothing more than "its own thing".

2.2 Single-paradigm Optical computing

An optical computer [6, 8] is a physical information processing device that uses photons to transport data from one memory location to another, and processes the data while it is in this form. In contrast, a conventional digital electronic computer uses electrons (travelling along conductive paths) for this task. The optical data paths in an optical computer are effected by refraction (such as the action of a lens) or reflection (such as the action of a mirror).

The advantages [8] of optical computing over electronic computing include: data paths that can intersect and even completely overlap without corrupting the data in either path (allowing highly parallelised computations and greatly simplifying circuit design); the ability to encode a two-dimensional spatial function in the cross-section of a single beam of light; low energy consumption (an argument deriving from the fact that optical computers in principle generate very little heat).

Efforts to exploit the tightly-coupled parallelism afforded by optics have largely focussed on forms of correlation for pattern recognition [5, 18, 19] and applications of optical matrix multiplication [2, 3, 9, 20]. However, optical computing has been hampered by the fact that a switch or branch instruction is difficult to implement in optics. Furthermore, currently there is no convincing alternative to using electronic devices as input (liquid crystal display panels, for example) and output devices (digital cameras, for example) for optical computers. Finally, the optical computation in such optoelectronic implementations have been effected in only single pass convolutions and products. These optical computations are not general purpose, and not Turing-equivalent.

Since the 1990s with optical computing we have been stuck with only two possibilities. We have the unconventional optical computers on one hand that are not general-purpose, and the general-purpose optical computers [4] on the other hand that admit none of the computational complexity advantages of unconventional computation. It is not surprising that the fortunes of optical computing, once so highly promising, have floundered. However, as explained in §4, heterotic unconventional computing can bypass this deadlock.

2.3 Single-paradigm Bacterial computing

Bacteria are able to adapt to and explore complex environments by coupling their information processing molecular systems to the production of specific proteins (including fluorescent proteins) and to the activation of powerful molecular motors (flagella) that propel cells forward or make them change direction.

Sophisticated molecular techniques have been developed recently to genetically modify bacteria. These techniques allow biologists to isolate molecular components from different bacteria, reassemble them into new biomolecular systems and insert them into other bacteria producing new bacterial strains.

In order to guide this complex process the use of computational modelling has become crucial. In this respect, a new discipline called Synthetic biology is emerging. Synthetic biology integrates traditionally compartmentalised disciplines such as genetic engineering, molecular biology, protein design, control theory and computational modelling. Synthetic biology seeks the design and implementation of new biomolecular components, devices and systems not available in living organisms as well as the redesign of already existing molecular systems aiming at the achievement of beneficial phenotypes to the human being. This discipline exploits the recent advances in the chemical synthesis of DNA that allows scientists to obtain novel designer DNA sequences and the application engineering methodologies such as the characterisation and standardisation of components and the hierarchical, modular and parsimonious design of circuits using computational methods.

Following the methodologies proposed by Synthetic biology computational devices such as inverters, oscillators and logic gates have been implemented in bacteria using mainly chemical signals. However, the wiring problem is serious: how to connect up the inputs and outputs of these components to produce a larger circuit. Most suggested approaches do not scale beyond a few tens of components, and do not allow long distance communication.

2.4 Single-paradigm Chemical computing

Among different types of chemical substrates performing information processing tasks spatially distributed excitable and oscillatory media look especially interesting: excitable chemical reactions are responsible for information processing in a nerve system of living organisms.

The time evolution of a distributed chemical medium can be described by the reaction-diffusion equation, so the field of research is named reaction-diffusion computing [1]. The Belousov-Zhabotinsky (BZ) reaction has been studied for more than 50 years, and it gives an experimental background for experimental investigations of chemical computing. Many recent studies on information processing with BZ reaction information have been concerned with excitation spikes (defined as a peak of concentration of the catalyst in its oxidized form) travelling in the medium.

The simplest and most commonly used translation of chemical evolution into the language of computer science is based on a Boolean representation: the presence of excitation pulse at a selected point of space is associated with logical TRUE and the absence of excitation with logical FALSE. Within such a representation we can construct the basic binary logical gates, memory cells or frequency filters with a nonlinear chemical medium [21]. Research on information processing with nonlinear media has shown that the geometry of the medium plays an equally important role as the chemical dynamics. Many recent studies are concerned with a structured medium, for example in the form of droplets containing solution of reagents [16].

There are photosensitive variants of the BZ reaction. Such reactions are important because illumination gives an additional control parameter that can be applied in an experiment.

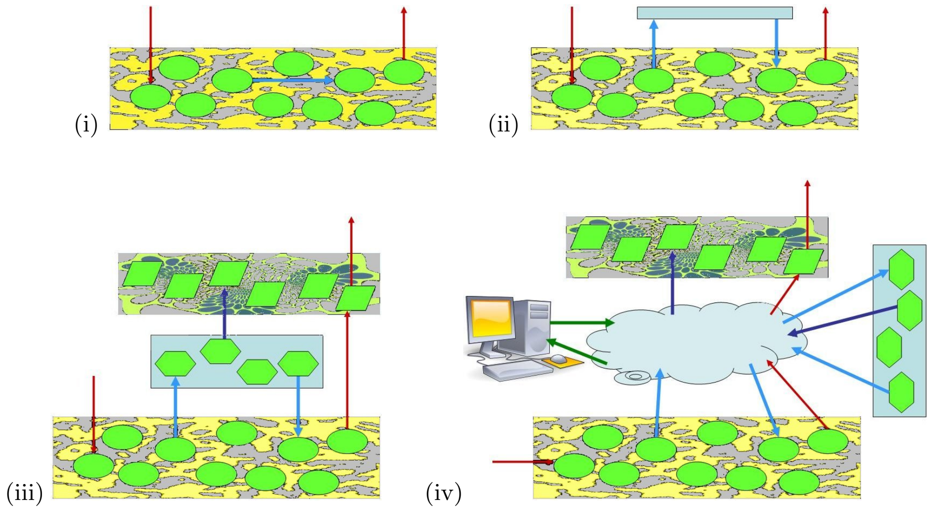

Fig. 1. Conceptual overview of the stages in moving to a full heterotic computational system: (i) a single unconventional computer, with internal communication; (ii) a minimal heterotic system, with communication implemented by an external layer; (iii) a full heterotic computational system with multiple communicating computational layers; (iv) a full heterotic computational system, with the components communicating via an API.

Chemical computing also suffers from the wiring problem. In a single substrate system, communication between droplets happens within the chemical substrate, by diffusion of reagents through the lipid layers of contacting droplets, and so is restricted to nearest neighbours.

3 Heterotic computing

3.1 A layered heterotic architecture

The fact that several existing unconventional computational systems have similar layered structures was first noted in [7]; a further case was identified in [15]. Several of these are quantum computing architectures, but this is not a necessary component for heterotic computation. We have used these examples as inspiration for our heterotic unconventional computation architecture [7] comprising diverse layers of communicating computational systems (*heterotic* is an adjective borrowed from genetics, where it means "hybrid vigour"). These existing cases demonstrate that diverse computational layers can be combined in a way that respects the intrinsic computational power of each, whilst resulting in a system more powerful than the sum of its parts. We dub this gain in power "the heterotic advantage".

Conceptually, we have a progression of hybrid architectures to the full heterotic system, as outlined in figure 1. This progression is:

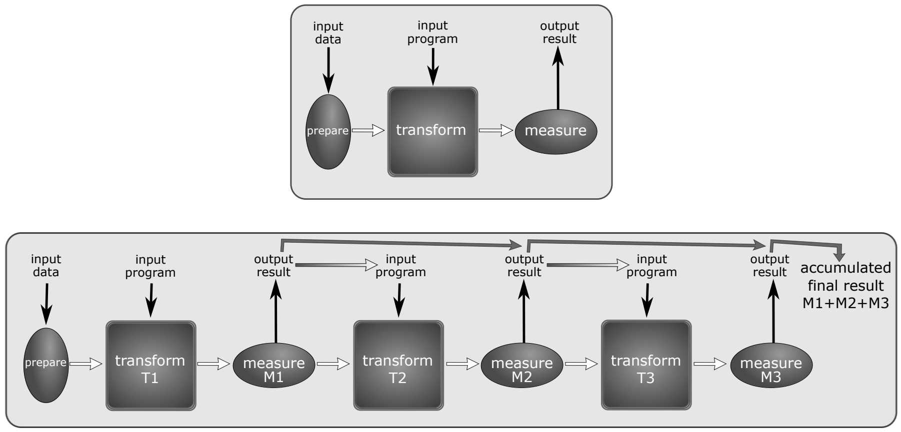

Fig. 2. Illustrations of the concepts of state preparation, transformation, and measurement, in a single shot (upper) and multi-step with feedback (lower). These are shown with a user/control layer on top of the computational substrate, to emphasise that even single paradigms have elements of hybrid computation in their structure and operation.

1. A single unconventional computer, where the internal components (represented by ellipses in figure 1) use a diffusive communication medium, and so have difficulty establishing targetted long-range internal communication (internal arrow). However, the components can receive inputs from and generate outputs to the outside world.
2. The internal communication is instead implemented by a second "communication" layer, resulting in a minimal heterotic system (the communication layer is doing minimal computation itself).
3. A full heterotic computational system: each layer is a particular computational subsystem, communicating with other computational layers.
4. A full heterotic computational system, with the components communicating via an API (embodied, or explicit, depending on particular communication path).

Basic notions of our heterotic architecture include: state preparation, transformation and measurement, common to classical computation and single paradigm unconventional computation as well as to our heterotic model (figure 2). The combination of heterogeneous sub-systems and the information flows between them is itself a physical process, which needs to be modelled, at a suitable level of abstraction (figure 3).

3.2 Distinctive features of heterotic computing

There are three features of heterotic computation that collectively tend to make it distinctive. These are: multi-paradigm compositionality, heteroticity, and physical embodiment.

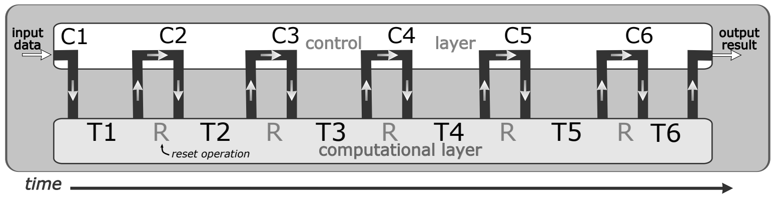

Fig. 3. The amount of memory in the control layer is a key parameter. The computational layer can either reset (R) between transformations (T), or continue to evolve. If R is the identity, so it preserves the state from T_i to T_{i+1}, then the computational layer stores its full state after output as input to the next transformation. If not, it is limited by the amount of memory in the control layer to pass information between transformations.

Multi-paradigm compositionality. The essence of heterotic computation is to combine different computational paradigms, with widely differing physical realisations. This must be done in a robust and tractable fashion.

Heteroticity. Not only are the systems hybrid in nature, but one of the key issues is to identify and characterise computational advantages which may arise from this hybridity. This is the issue of the "heterotic advantage".

Physical embodiment. Different physical substrates have their own, widely varying characteristics. It is necessary to find a common level of description, and the right level of abstraction, which allows realistic models for gauging computational expressiveness and cost, while still allowing for a reasonable degree of generality.

3.3 Addressing single-substrate issues

Heterotic unconventional computation addresses single-substrate issues in two ways:

1. The **richer approach** of allowing the computational system to include multiple unconventional devices, rather than struggling to perform all computation in a single system, can solve the problems highlighted above:
   - it can solve the non-universality problem (if this is indeed a problem in a given case) by combining non-universal special-purpose sub-systems into a universal whole
   - it can solve the wiring problem by allowing long-distance targetted communication to take place in a different (wiring) layer
2. A heterotic **semantic framework**, which should provide
   - a foundation for composing systems, and defining their computation
   - tools for analysing the computational power of the combination, including the information encoding issues
   - tools for analysing the computational power involved in i/o transduction (from its explicit focus on the communication between layers)

3.4 New issues with the heterotic approach

The richer approach of combining multiple systems does raise new issues, for example:

- Widely different timescales between layers, ranging from nanoseconds and less for optics, to hours for bacteria.
- The requirement for signal transduction between systems using different information embodiment or encodings. Classical computational transduction in terms of changing medium is required mostly only between the computation and the outside world (although there can be much transduction concerned with converting encodings). Note that use of an appropriate unconventional layer may remove or simplify the transduction requirement with the outside world, if the layer computes directly in the substrate of the outside world (for example, processing chemical signals).

A semantic framework should allow these issues to be exposed and analysed, for example, ensuring that no computation is "hidden" in the transduction process.

4 Examples of heterotic computing

In this section, we demonstrate how heterotic computers can allow computational problems to be addressed in a more "natural" way than in a single substrate alone, by describing three hypothetical examples. These cover a broad spectrum of substrates (physical, chemical, biological) with a corresponding spectrum of lengthscales (nano to millimetre scale), timescales (nanoseconds to hours), theoretical bases (fully characterised to phenomenological), and noise regimes. Any semantic framework for heterotic computing would have to encompass all these spectra.

4.1 Heterotic Optical-Bacterial computing

This section describes how a heterotic computer comprising an optical layer and a bacterial layer can be used to address the "wiring problem" in bacterial computing. This wiring problem is how to join computation outputs from one subsystem to inputs to another subsystem within the bacterial computer. In an Optical-Bacterial computer, these communication "wires" are implemented in the optical layer.

This approach could be used to implement optical control of phototactic bacteria in the following way. There are photoswitchable biomolecular components in bacteria that respond reversibly to red and to green light wavelengths. When these components are activated by the relevant wavelengths, they bind to specific sequences of DNA, initiating the production of whatever proteins are coded by the genes fused to these promoters [17]. These expressed proteins could be gene-engineered to be green and red fluorescent proteins, and proteins that activate a flagella motor to exhibit positive or negative phototaxis.

The optics is used to provide input to the bacteria (with potentially different inputs to bacteria in different spatial regions, allowing a 2D encoding of the input). The inputs cause phototaxis: the bacteria move in an input-dependent way. This movement comprises the bacterial computation. The inputs also cause fluorescence. The optical system is then used to detect and read out the new locations of the moved bacteria (that is, to read the result of the bacterial computation). This provides one input-to-output iteration of the computation.

Bacteria can communicate indirectly through this optical process: the outputs, encoded in the movement and fluorescence, can be input to a different, potentially remote, part of the bacterial system by means of the optical system.

4.2 Heterotic Chemical-Optical computing

This section describes how a heterotic computer comprising an optical layer and a chemical layer can be used to address the control flow problem in optical computing, with a different model of encoding information in a chemical system.

Consider a geometrically restricted chemical medium like, for example, a droplet containing the solution of reagents of an oscillatory BZ reaction. The size of the droplet objects and the kinetics of the reaction define stable spatio-temporal structures that can appear in a restricted geometry. The kinetics can be optically controlled where the reaction is photosensitive. An object with a stable spatio-temporal structure can be considered as a memory cell with a state defined by that structure.

A droplet has two obvious states: chemical oscillations are present, or they are absent. However, optical techniques can be used to identify and classify the spatio-temporal structure into further distinct states, to make a memory cell with a capacity larger than one bit. Suitable optical perturbation of the medium can then change one spatio-temporal structure into another, allowing us to write to memory.

A single droplet can work as a memory; further functionality can obtained from a number of droplets, with some selected as input and others as output. The result of such system is defined as the state of the output object(s) at a given time after the states of input objects are defined. A collection of such droplets forms our chemical computational layer.

Using an optical computation layer, optical control/feedback can be implemented between droplets separated by long distances, which significantly extends the number of operations that can be performed by a fixed, limited number of droplets.

4.3 Heterotic Bacterial-Chemical computing

This section describes how a heterotic computer comprising a bacterial layer and a chemical layer can be used to implement a novel reaction-diffusion paradigm computation, in bacteria communicating via a chemical layer, rather than in chemicals alone.

It is well known that a chemical excitable medium can process information coded in excitation pulses. This excitability can be induced by specific chemical signals in bacterial populations. The medium comprises spatially distributed bacteria that have been genetically modified in order to behave as an excitable medium. For example, molecular components from the quorum sensing systems of the bacteria could be used. Circuits can be designed to make bacteria synthesise more quorum-sensing molecules (called autoinducer homoserine lactones, or AHLs) when they sense them, in order to propagate the excitation across populations of bacteria. Then, after a delay, the sensing of AHL can be programmed to activate the production of repressor proteins and AHL-degrading proteins that will stop the synthesis of AHL and remove all the remaining AHL, making the bacteria return to their initial unexcited state. Different genetic circuits can provide bacteria with specific properties such as refractory time and activation threshold.

By also including chemotactic abilities in the bacteria, they can be induced to move as well as oscillate, allowing the possibility of controlled re-programming of the reaction-diffusion system.

4.4 Heterotic Optical-Bacterial-Chemical computing

Clearly several of these techniques could be combined to form a three layer heterotic computer, with a combination of mobile (phototactic and chemotactic) fluorescing bacteria and chemical droplets performing a computation controlled by optical and chemical signals.

If optics is used for some computation as well as communication, the inputs and outputs to the optical layer are effected by biological and chemical substrates, avoiding the need for some, if not all, fundamentally serial optoelectronic devices. The control flow branching that is difficult in optical imaging can also be performed efficiently in these substrates.

5 Discussion and conclusions

We have described issues with single substrate unconventional computing, and a heterotic computational architecture for overcoming these. We have illustrated this architecture with a range of possible two-substrate and three-substrate heterotic systems, which permit novel approaches to the wiring problem, information encoding, and other issues. Clearly this approach can be extended to more computational layers, and more kinds of layers, including quantum computational layers, and also classical computational layers. A quantum layer, for example, can exist in the same physical substrate as another computational layer. This illustrates the fact that additional computational layers are not necessarily additional physical substrates, but may also be just different computational models of a given physical substrate.

Heterotic unconventional computation enables the most effective and efficient devices to be applied to a wide range of specific problems, by combining multiple different kinds of unconventional substrate, each exploiting their individual strengths, and overcoming their individual weaknesses. This approach can make unconventional computing a more mature science, and open up a route to incorporating it in mainstream technology, thereby allowing it to fulfil its promised potential contribution to a future of pervasive, ubiquitous, embodied computing.

Acknowledgements

VK is supported by a UK Royal Society University Research Fellowship. TJN acknowledges support from Science Foundation Ireland under the National Development Plan. MJPJ acknowledges funding from grant no TIN2009-13192; FJRC acknowledges funding from grant no JCI-2010-06532.

References

1. Adamatzky, A., Costello, B.D.L., Asai, T.: Reaction-Diffusion Computers. Elsevier (2005)
2. Caulfield, H.J., Kinser, J.M., Rogers, S.K.: Optical neural networks. Proceedings of the IEEE 77, 1573–1582 (1989)
3. Farhat, N.H., Psaltis, D.: New approach to optical information processing based on the Hopfield model. Journal of the Optical Society of America A 1, 1296 (1984)
4. Huang, A.: Architectural considerations involved in the design of an optical digital computer. Proceedings of the IEEE 72(7), 780–786 (1984)
5. Javidi, B.: Nonlinear joint power spectrum based optical correlation. Applied Optics 28(12), 2358–2367 (1989)
6. Karim, M.A., Awwal, A.A.S.: Optical Computing: An Introduction. Wiley (1992)
7. Kendon, V., Sebald, A., Stepney, S., Bechmann, M., Hines, P., Wagner, R.C.: Heterotic computing. In: Unconventional Computation 2011. LNCS, vol. 6714, pp. 113–124. Springer (2011)
8. Naughton, T.J., Woods, D.: Optical computing (invited). In: Meyers, R.A. (ed.) Encyclopedia of Complexity and Systems Science. pp. 6388–6407. Springer (2009)
9. Naughton, T., Javadpour, Z., Keating, J., Klíma, M., Rott, J.: General-purpose acousto-optic connectionist processor. Optical Engineering 38(7), 1170–1177 (1999)
10. Stepney, S.: The neglected pillar of material computation. Physica D: Nonlinear Phenomena 237(9), 1157–1164 (July 2008)
11. Stepney, S.: Nonclassical computation: a dynamical systems perspective. In: Rozenberg, G., Bäck, T., Kok, J.N. (eds.) Handbook of Natural Computing, volume II, chap. 52. Springer (2011)
12. Stepney, S., Abramsky, S., Adamatzky, A., Johnson, C., Timmis, J.: Grand challenge 7: Journeys in non-classical computation. In: Visions of Computer Science, London, UK, September 2008. pp. 407–421. BCS (2008)
13. Stepney, S., Braunstein, S.L., Clark, J.A., Tyrrell, A., Adamatzky, A., Smith, R.E., Addis, T., Johnson, C., Timmis, J., Welch, P., Milner, R., Partridge, D.: Journeys in non-classical computation I: A grand challenge for computing research. International Journal of Parallel, Emergent and Distributed Systems 20(1), 5–19 (Mar 2005)
14. Stepney, S., Braunstein, S.L., Clark, J.A., Tyrrell, A., Adamatzky, A., Smith, R.E., Addis, T., Johnson, C., Timmis, J., Welch, P., Milner, R., Partridge, D.: Journeys in non-classical computation II: initial journeys and waypoints. International Journal of Parallel, Emergent and Distributed Systems 21(2), 97–125 (Apr 2006)

15. Stepney, S., Kendon, V., Hines, P., Sebald, A.: A framework for heterotic computing. In: 8th workshop on Quantum Physics and Logic (QPL 2011). EPTCS (2011)
16. Szymanski, J., Gorecka, J.N., Igarashi, Y., Gizynski, K., Gorecki, J., Zauner, K.P., de Planque, M.: Droplets with information processing ability. International J. Unconventional Computing 7, 141–158 (2011)
17. Tabor, J.J., Levskaya, A., Voigt, C.A.: Multichromatic control of gene expression in *Escherichia coli*. Journal of Molecular Biology 405(2), 315–324 (2011)
18. VanderLugt, A.: Signal detection by complex spatial filtering. IEEE Transactions on Information Theory 10(2), 139–145 (1964)
19. Weaver, C.S., Goodman, J.W.: A technique for optically convolving two functions. Applied Optics 5(7), 1248–1249 (1966)
20. Woods, D., Naughton, T.J.: Optical computing: Photonic neural networks. Nature Physics 8(4), 257–259 (2012)
21. Yoshikawa, K., Motoike, I.N., Ichino, T., Yamaguchi, T., Igarashi, Y., Gorecki, J., Gorecka, J.N.: Basic information processing operations with pulses of excitation in a reaction-diffusion system. International J. Unconventional Computing 5(1), 3–37 (2009)