

To appear in the *International Journal of General Systems*
Vol. 00, No. 00, Month 201X, 1–14

Population-based microbial computing: A third wave of synthetic biology?

Martyn Amos*

*School of Computing, Mathematics and Digital Technology, Manchester Metropolitan
University, Manchester M1 5GD, United Kingdom.*

(Received 00 Month 201X; final version received 00 Month 201X)

Synthetic biology is an emerging research field, in which engineering principles are applied to natural, living systems. A major goal of synthetic biology is to harness the inherent “biological nanotechnology” of living cells for the purposes of computation, production, or diagnosis. As the field evolves, it is gradually developing from a single-cell approach (akin to using standalone computers) to a distributed, *population-based* approach (akin to using *networks* of connected machines). We anticipate this eventually representing the “third wave” of synthetic biology (the first two waves being the emergence of *modules* and *systems* respectively, with the latter still yet to peak). In this paper we review the developments that are leading to this third wave, and describe some of the existing scientific and technological challenges.

Keywords: synthetic biology; distributed computing; unconventional computing.

1. Introduction

Although the concept of “synthetic biology” dates back to the early 1900s, with the publication of Leduc’s *La Biologie Synthétique*, the first contemporary definitions of the term came in the the 1970s, shortly after the discovery of restriction enzymes and the development of recombinant DNA:

“Up to now we are working on the descriptive phase of molecular biology. . . . But the real challenge will start when we enter the synthetic phase of research in our field. We will then devise new control elements and add these new modules to the existing genomes or build up wholly new genomes. This would be a field with an unlimited expansion potential and hardly any limitations to building “new better control circuits” or . . . finally other “synthetic” organisms, like a “new better mouse” . . . I am not concerned that we will run out of exciting and novel ideas . . .” (Szybalski, 1974).

In this single paragraph (delivered as part of his contribution to a conference discussion panel) Szybalski effectively described a “manifesto” for a developing field that is only now beginning to demonstrate its true potential. *Synthetic biology* is an emerging research area that currently defies a single, agreed definition (for discussions of the history of the term, see Benner & Sismour (2005); Serrano

*Email: M.Amos@mmu.ac.uk

(2007)). Synthetic biology has been referred to as “...the engineering-driven building of increasingly complex biological entities for novel applications” (Heinemann & Panke, 2006), or “...the science of selectively altering the genes of organisms to make them do things that they wouldn’t do in their original, natural, untouched state” (Church & Regis, 2012), while others have labelled the term “premature, even boastful” (Bray, 2009).

In terms of specific research goals, some partition the field into two main camps (which are by no means disjoint): those who wish to understand life and its underlying processes, and those who wish to harness engineered living systems for specific applications. As Benner and Sismour argue, “Synthetic biologists come in two broad classes. One uses unnatural molecules to reproduce emergent behaviours from natural biology, with the goal of creating artificial life. The other seeks interchangeable parts from natural biology to assemble into systems that function unnaturally. Either way, a synthetic goal forces scientists to cross uncharted ground to encounter and solve problems that are not easily encountered through analysis. This drives the emergence of new paradigms in ways that analysis cannot easily do. Synthetic biology has generated diagnostic tools that improve the care of patients with infectious diseases, as well as devices that oscillate, creep and play tic-tac-toe.” (Benner & Sismour, 2005).

In this review we mainly consider work undertaken by the *second* class of synthetic biologists; those who seek to engineer existing biological systems to achieve “unnatural” results (for coverage of related research into the creation of so-called “artificial life”, the reader is directed to Forster & Church (2006); Noireaux & Libchaber (2004); Pohorille & Deamer (2002); Rasmussen, et al. (2003)). Such results may include the production and/or delivery of drugs or therapeutic agents, pollution clean-up, pathogen detection, or micro-scale fabrication (for discussion of possible applications of synthetic biology, see Khalil & Collins (2010); McDaniel & Weiss (2005)).

For many, synthetic biology is a natural progression of “traditional” genetic engineering. As Purnick & Weiss (2009) observe, “Conventional genetic engineering approaches for solving complex problems typically focus on tweaking one or a few genes. Synthetic biology, by contrast, approaches these problems from a novel, engineering-driven perspective that focuses on wholesale changes to existing cellular architectures and the construction of elaborate systems from the ground up. . . . Whether addressing an existing problem or creating new capabilities, effective solutions can be inspired by, but need not mimic, natural biological processes. Our new designs can potentially be more robust or efficient than systems that have been fashioned by evolution.”

In the same paper, Purnick & Weiss (2009) introduce the notion of a “second wave of synthetic biology”, in which small modules are pieced together into larger systems. We adopt this narrative device here, as it also offers a useful over-arching framework for capturing what followed in subsequent years. Usefully, the authors also examine the synthetic biology literature (between 2000 and 2009) in order to establish the maximum and average complexity (measured in terms of the number of promoter regions) of published complete synthetic biological circuits. Between years 2000 and 2003, maximum complexity remained fairly constant; it then increased by 50% between 2003-2006, and plateaued between 2006-2009. We therefore argue that the “first wave” of synthetic biology may be naturally assigned to the period 2000-2003, with the “second wave” beginning in 2003. In what follows, we describe each wave in turn, using an archetypal example from the literature for the purposes of illustration.

The paper is organized as follows: we begin with a review of early work on synthetic biology, which dealt mainly with the construction of fundamental *components* or *modules*. We then describe the second wave of research in the field, in which modules were pieced together into larger *systems*. Although “second wave” research is still ongoing, we anticipate the emergence of a growing “third wave”, in which researchers connect collections of *heterogenous* engineered systems into large-scale, population-based solutions, or *networks*. We conclude with a discussion of outstanding issues, and consider areas for future research.

2. The first wave: “modules”

“Natural organisms are, as a rule, much more complicated and subtle, and therefore much less well understood in detail, than are artificial automata. Nevertheless, some regularities which we observe in the organisation of the former may be quite instructive in our thinking and planning of the latter.” (von Neumann, 1951).

With the publication of his seminal work on self-reproducing automata, von Neumann laid the foundations of the field that would become known as “artificial life” (Levy, 1992). Like Descartes and Hobbes before him, John von Neumann fundamentally believed that life was based on *logical* principles. This was echoed by the Nobel Laureate Jacques Monod, when he argued that “The logic of biological regulatory systems abides, . . . like the workings of computers, by the propositional algebra of George Boole.” (Monod, 1971).

For a review of the “computational” view of nature, and how it has informed the development of synthetic biology, the reader is referred to Amos (2006). In terms of their construction, both computers and biological systems may be arranged in a hierarchy of layers, with each layer corresponding to a specific level of organization. At the base of the hierarchy we have the simplest components, which form the initial building blocks for subsequent layers. As we move from layer to layer, in a “bottom up” fashion, the system grows in complexity, but in an inherently *modular* fashion (Wall, et al., 2004). Andrianantoandro, et al. (2006) make this link explicit, and in Figure 1, we present an augmented version of their scheme.

At the lowest level of computer architecture (the *physical layer*), we have fundamental *components*, such as transistors and resistors. In synthetic biology, these components correspond to basic elements such as *promoters* (regions of DNA that initiate the transcription of a gene) and *repressors* (proteins which inhibit gene transcription (Purnick & Weiss, 2009)). These may then be connected to form *devices*, such as Boolean logic gates (akin to biochemical reactions in synthetic biology). In the “first wave” of synthetic biology, these devices were combined to form *modules* to achieve specific tasks, such as *switching*, *pulse generation* and *oscillation*. For the purposes of illustration, we now focus on the last of these.

In physics, an oscillator is a system that produces a regular, periodic “output”. Familiar examples include a pendulum or a vibrating string. Linking several oscillators together in some way gives rise to *synchrony* – for example, heart cells repeatedly firing in unison, or millions of fireflies blinking on and off, seemingly as one (Strogatz, 2003), and the study of oscillations in *networks* can shed light on their structure (and, thus, their function) (Dorrian, et al., 2013).

Oscillators are fundamental to biology, but they are also of interest to engineers, since they form the basis for *counting* (and engineered synchronisation). Synthetic biology combines both disciplines, so the construction of oscillators within living cells was a natural early step. In the early 1960s, Brian Goodwin described a mathematical model of a simple biological oscillator, whereby a specific protein

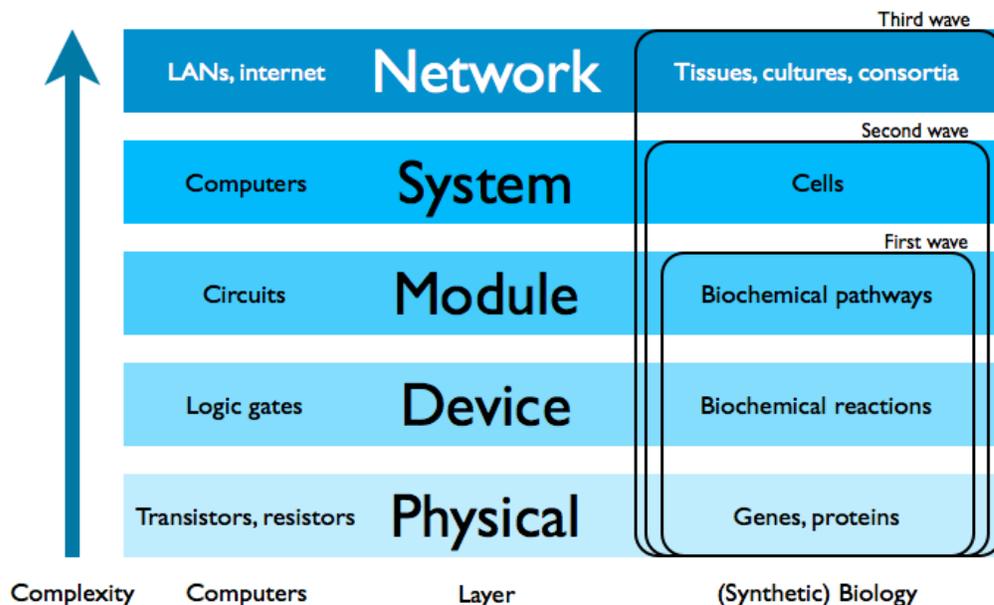


Figure 1. Hierarchical mapping between existing layers of abstraction for both computer engineering and synthetic biology, showing both the mapping between them, and the “waves” of activity in the latter area (adapted from Andrianantoandro, et al. (2006)).

would inhibit transcription of the gene responsible for its production (Goodwin, 1963; Ruoff, et al., 1999). An important feature of this theoretical oscillator is the fact that it uses *negative feedback*; *stability* (defined in this context as “the tendency of a system to remain close to a steady state” (Hasty et al., 2002)) is crucial to the “quality” of the oscillation (that is, the range of fluctuation of component concentrations over time), and engineered networks that use negative feedback are demonstrably more stable than their unregulated counterparts (Becskei & Serrano, 2000).

The utility of negative feedback was highlighted when Elowitz & Leibler (2000) demonstrated their “repressilator”; an oscillator based on a circuit of gene transcription *repressors*. This is the archetypal “first wave” module. An abstract version of their scheme is depicted in Figure 2; essentially, the repressilator is constructed from three genes connected in a cycle, such that each gene is repressed (turned “off”) by its predecessor, and, in turn, represses its successor. Gene 3 is additionally connected to a *reporter* gene (which causes cells to “glow”), represented by a light-bulb, so that the bulb is “on” if and only if Gene 3 is “on”.

We assume that the default state of all three genes is “on”; that is, a gene that is not actively being turned “off” (repressed) will default to “on”. All genes start in the “off” position, via mutual repression. In order to “initialise” the system, we temporarily disrupt the repressive connection between Gene 1 and Gene 2. This allows Gene 2 to turn “on”, but, more importantly, it also causes Gene 3 to be turned “off” (Figure 2(a)). In turn, this removes the repressive connection between Gene 3 and Gene 1, allowing Gene 1 to default back to “on” (Figure 2(b)). Gene 1 then turns “off” Gene 2, which can no longer repress Gene 3, so Gene 3 turns “on”, and the light goes “on. Gene 3 then turns “off” Gene 1 (Figure 2(c), allowing Gene 2 to come back “on”. This, in turn, turns “off” Gene 3, and the cycle continues.

Importantly, the transcriptional repressors selected by Elowitz and Leibler are not found in any naturally-occurring oscillator (so they could be sure that they were not taking advantage of evolved behaviour, and that they had constructed a genuinely new system). The appropriate genes were “inserted” into *E. coli* bac-

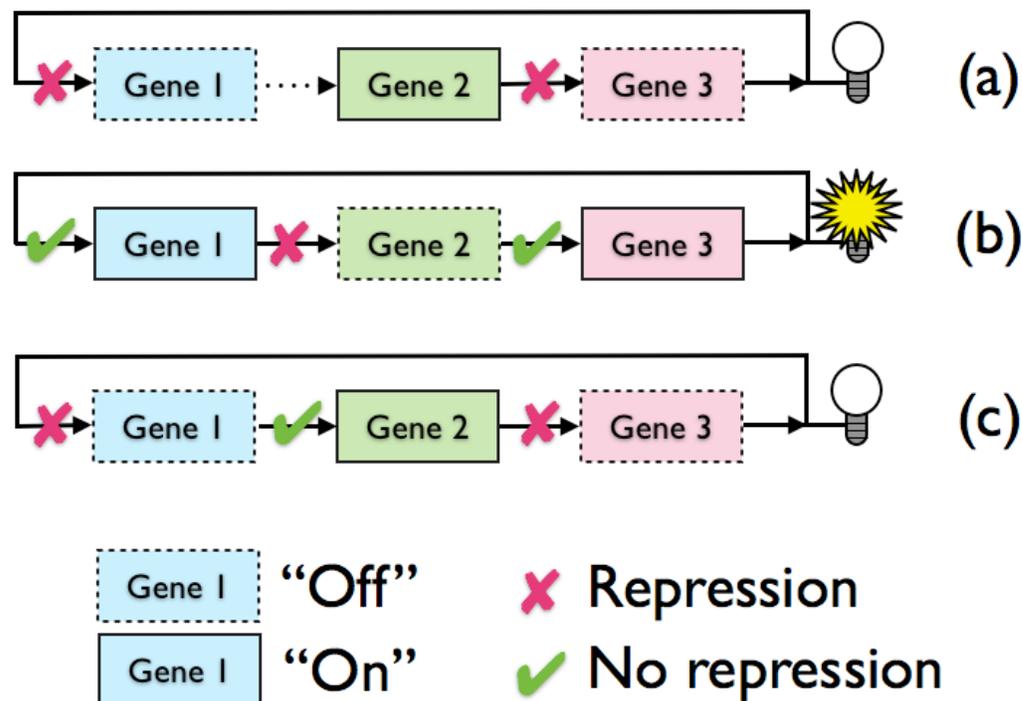


Figure 2. Schematic representation of the repressilator. (a) Initial disruption of Gene 1's repression, allowing Gene 2 to turn "on" and repress Gene 3. (b) Gene 1 defaults to "on" in absence of repression from Gene 3, thus turning "off" Gene 2, and allowing Gene 3 to turn "on" (lighting the bulb). (c) Repression of Gene 1 by active Gene 3, returning system to the beginning of the cycle.

teria, and the system "jump-started" by temporary disruption of the first gene's repression signal. Oscillatory behaviour (measured in terms of the intensity fluorescence) was observed at the level of individual cells. Importantly, the mean period of oscillation (around 160 minutes) was *larger* than the typical cell-division time (50-70 minutes), meaning that the *state* of the oscillation network was preserved from one generation of cells to the next.

This was a landmark result in "first wave" synthetic biology, because it demonstrated the feasibility of integrating, in a rational, engineered manner, a *number* of fundamental components (i.e., repressors, promoters) together into a single module (as opposed to making modifications to existing structures, or inserting single genes). Once these modules were constructed and characterised, the aim was to then use them to populate DNA-based "libraries", in exactly the same way that software developers draw on pre-existing code modules with a well-defined interface and behaviour. New modules quickly followed; around the same time that the repressilator work appeared, a number of other important "first wave" experiments were published.

In the same issue of *Nature*, Gardner, et al. (2000) demonstrated a genetic "toggle switch", which allowed external control of a bi-stable (two state) circuit engineered into *E. coli*. They observed that toggle switching had not yet been "demonstrated in networks of non-specialised regulatory components." That is to say, at that point nobody had been able to construct a toggle switch using genes that hadn't already been selected by evolutionary processes to perform that specific task. As with the repressilator, Gardner and colleagues took an inherently "bottom-up" approach to building their circuit; rather than taking an existing circuit and then making changes (i.e., in a top-down fashion), they first constructed a mathematical model of a synthetic, bistable gene circuit. They chose two genes that were mutually in-

hibitory - that is, each produced a molecule that would turn the other off. The important thing to note about their system is that it did not have one single input; although the authors acknowledged that bistability *might* be possible using only a single promoter, they anticipated problems with robustness and tunability. Instead, they selected a system whereby each “side” of the switch could be “pressed” by a different stimulus - the presence of a chemical on one, and a change in temperature on the other. If the system was in the state induced by the chemical, it would stay in that state until the *temperature* was changed, and would only change back again if the chemical was reintroduced (and vice versa). Importantly, these stimuli did not have to be applied continuously - a short burst was sufficient to cause the switch to flip over. The authors anticipated several possible applications for their module, borrowing from the language of computing: “As a practical device, the toggle switch . . . may find applications in gene therapy and biotechnology . . . Finally, as a cellular memory unit, the toggle forms the basis for ‘genetic applets’ - self-contained, programmable, synthetic gene circuits for the control of cell function.” Gardner, et al. (2000)

As Hasty and colleagues note, *mathematical modelling* was fundamental to the success of both projects: “As in the case of the toggle switch, a mathematical model was instrumental in the process of designing the repressilator. Although the ring network architecture is theoretically capable of sustaining oscillations, not all parameter choices give rise to oscillatory solutions. The modelling work indicated that oscillations were favoured by high protein synthesis and degradation rates, large cooperative binding effects, and efficient repression. These theoretical conclusions led to specific design choices . . .” Hasty et al. (2002) As we will see, mathematical modelling and computational studies will continue to be central to the development of synthetic biology.

3. The second wave: “systems”

We now consider the “second wave” of the field, in which modules were brought together to form larger scale *systems*. As Likic, et al. (2010) point out (by highlighting the work of O’Malley & Dupré (2005) in the context of *systems biology*), the very notion of what constitutes a biological “system” can potentially be quite problematic. For our purposes, a synthetic biological system is a *set of modules* which interact to produce some coherent behaviour(s). These modules may be confined to a single cell, or they may be *distributed* over a number of different cells. We distinguish “systems” from the later “networks” by measuring their capacity for *information transmission*; generally, within systems, this is relatively low, whereas networks have much higher “bandwidth” capabilities.

A classic example of “second wave” synthetic biology concerns the engineering and connection of modules to achieve population-level coordination of behaviour (Basu, et al., 2005). The formation of *spatiotemporal patterns* is a fundamental property of many biological systems (Ball, 1999); synthetic biology can shed light on the underlying biological principles (Payne et al., 2013), but *engineered* pattern formation may also find significant future applications in tissue engineering or biological nanotechnology (Khalil & Collins, 2010). Controlling the way in which collections of cells interact to form specific patterns (or structures) is therefore of great interest.

Basu and colleagues harnessed the power of bacterial communication to effect differential spatial responses. Specifically, they used the bacterial *quorum sensing* (QS) mechanism (Atkinson & Williams, 2009), which is used by certain species

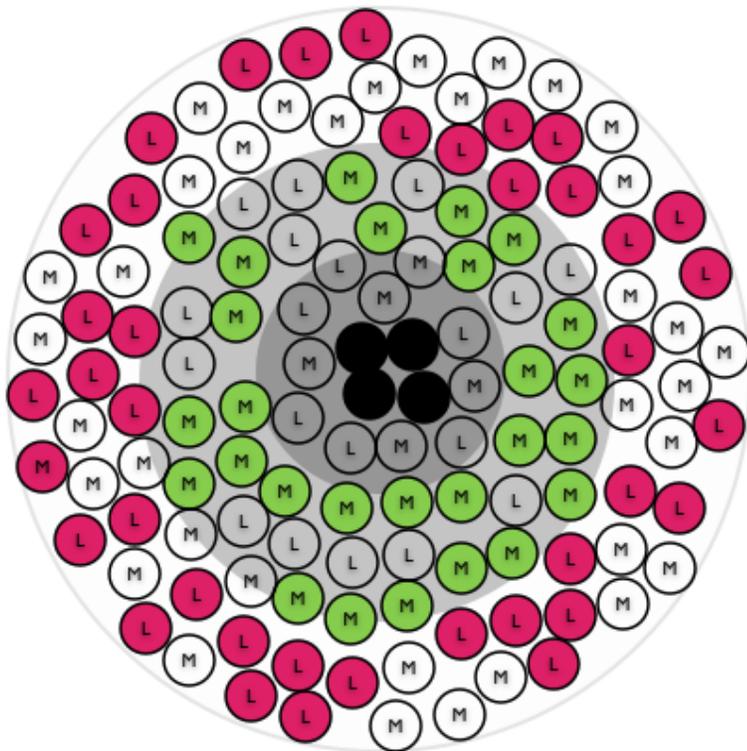


Figure 3. Schematic representation of the Basu, et al. (2005) programmed pattern generator. The petri dish contains black sender cells, and a randomly-distributed set of receiver cells, each labelled with the signal concentration to which they respond (either “L” for low, or “M” for medium). Signal concentration is represented by the nested circles (the darker the field, the higher the concentration). Close to the senders, where signal concentrations are high, no receiver cells respond. At an intermediate (“medium”) distance, only “M” receiver cells respond (labelled in green), and at a relatively far distance, only “L” (low concentration) receivers respond (red). An annular, “bullseye” structure forms in a decentralised, distributed fashion, purely as a result of differential responses to signal concentration levels.

to assess cellular concentrations, and thus coordinate gene expression (the idea being that a specific response, such as producing light, may only be “worthwhile” or feasible if there is a sufficient number - or *quorum* - of individuals present in a particular region). In QS, bacteria secrete *signalling molecules*, which they diffuse throughout the surrounding medium (with the concentration level obviously decaying with distance). They engineered several functional modules; one “sender” module, which was simply responsible for the production of signalling molecules, and several different “receiver” modules, which acted as *band-detectors* (that is, each would only produce a response when exposed to a relatively narrow range of signalling molecule concentrations). These modules were then inserted into *E. coli* bacteria; a schematic representation of the scheme is presented in Figure 3. Sender cells are represented in black, with receiver cells being labelled either “M” (they respond only to intermediate, or medium concentration levels) or “L” (respond only to low concentration levels). The sender cells are inoculated at the centre of the plate, and the receiver cells are randomly distributed across the surface. As the sender cells begin to produce the signalling molecule, the receivers respond *if and only if* the concentration level they detect matches their specific band. For example, an “L” receiver *will* respond if it is located relatively far from the senders, as concentration levels are low, but it will *not* respond if it is located relatively close to the senders, as the signalling molecule concentration will be too high.

In the Basu, et al. (2005) system, each receiver responded by producing a specific “coloured” fluorescent protein, which allowed the results to be easily visualised. The

authors observed clear “bullseye” patterns in the dish, which demonstrated the feasibility of engineering specific spatial patterns from a distributed, heterogenous population of cells containing engineered modules. Moreover, by placing a *number* of “clumps” of sender cells at specific points in space, they were able to obtain more complicated patterns (for example, a “clover leaf” pattern, formed using three different regions of senders).

Other notable examples of “second wave”, multiple module systems include a set of genetic clocks, synchronised using QS (Danino, et al., 2010), a logic evaluator that operates in mammalian cells (Rinaudo, et al., 2007), an edge detector (Tabor, et al., 2009), a synthetic predator-prey ecosystem (Balagaddé, et al., 2008) and a bistable (“push on, push off”) switch (Lou, et al., 2010). What all of these implementations have in common is the importance of computational/mathematical modelling, to either understand the behaviour of the system, find the best molecular sequences to use in specific situations, or to “fine tune” system parameters. However, engineering and computer science may also *inform* the development of the *next* wave of synthetic biology, through the adoption and application of basic principles. We consider this in the next section.

4. The third wave: “networks”

As Heinemann & Panke (2006) point out, the notion of *abstraction* (as used in its engineering context) is already fundamental to the field, but the *distributed computing* model will allow us to construct large-scale *networked* synthetic biology systems. In computer science, distributed systems are characterised by several features, including (1) *asynchrony* (that is, the lack of a global “clock”), (2) *local failure* of components, without global failure, and (3) *concurrency* (that is, components work in parallel) (Attiya & Welch, 2004). As biological systems share all of these features, the model may provide useful insights into future synthetic biology implementations. However, by their very nature of the underlying substrate (biological molecules and complexes, as opposed to silicon), engineered biological systems pose additional and significant challenges. As Purnick & Weiss (2009) highlight, “...engineering biological systems probably requires both new design principles and the simultaneous advance of scientific understanding. ... Beyond typical circuit design issues, synthetic biologists must also account for cell death, crosstalk, mutations, intracellular, intercellular and extracellular conditions, noise and other biological phenomena. A further difficult task is to correctly match suitable components in a designed system. As the number of system components grows, it becomes increasingly difficult to coordinate component inputs and outputs to produce the overall desired behaviour.” This theme is developed by Macía, et al. (2012), who point out that “In contrast to standard electronics, every wire needs to be a different molecule to properly connect different elements or cells. Inside a cell or in culture media the spatial insulation of wires that is assumed in electronics is no longer satisfied. As a consequence, the chemical diversity of constructs rapidly grows.”

It is clear that the internal environment of the cell places a natural upper bound on the complexity of engineered modules that can be introduced via synthetic biology techniques: “... inside a cell, the cables need to have a different implementation: different proteins must be used for each different pair. Additionally, because of the intrinsic difficulties of implementing them, the resulting constructs are usually specific for the given problem and cannot be reused afterwards” (Macía, et al., 2012).

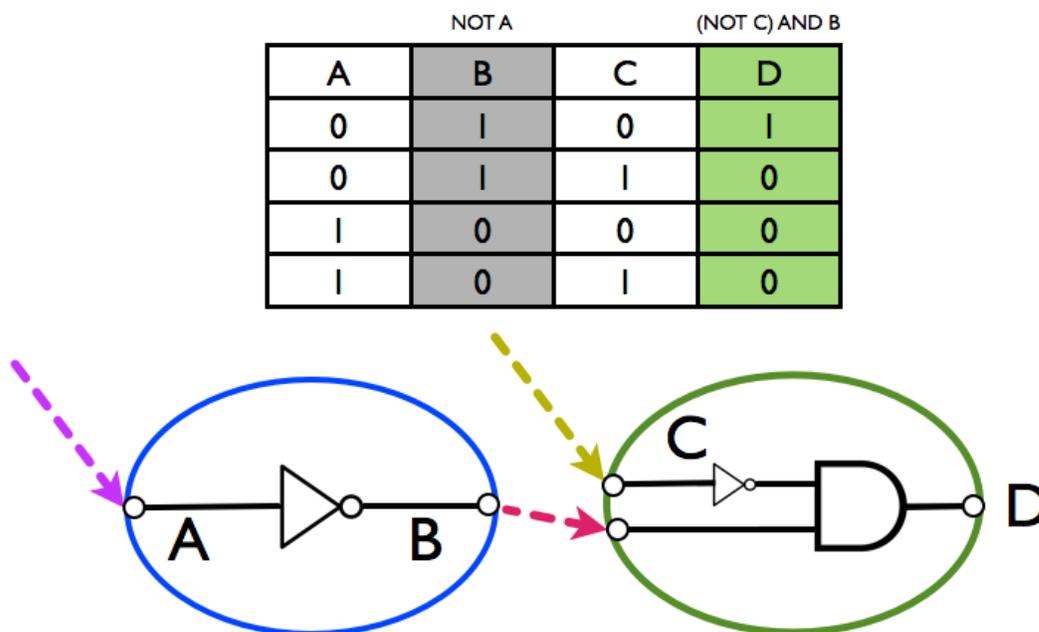


Figure 4. Schematic representation of the Regot, et al. (2011) multi-cellular approach to circuit evaluation of a NOR function. The left-hand cell is the NOT function from the library, and the right-hand cell is the N-IMPLY function (NOT C AND B) from the library. External inputs come from A and C, and B is used as a “wiring” molecule (that is, the output of the NOT function serves as one input to the N-IMPLY function). Taken together, these two cells implement, in a distributed fashion, the NOR function (with output denoted by D).

Attention has therefore turned to engineering solutions that go beyond single-cells, and consider microbial *consortia* (Brenner, et al., 2008; Solé & Macia, 2013). These are made up of *multiple* populations of microbes, that interact to give rise to behaviour that can be much more complex than the sum of the parts. Brenner, et al. (2008) list the benefits of using such mixed populations in synthetic biology: (1) The ability to perform complex tasks that are impossible for individual strains; (2) Robustness to environmental perturbation; (3) The ability to use communication to facilitate a division of labour; (4) Biological insight that can be derived from engineered consortia. These benefits map quite neatly onto the features of distributed computer systems listed above, and it is clear that engineered microbial consortia “... represent an important new frontier for synthetic biology.” (Brenner, et al., 2008).

In order to address the “wiring” issue, several approaches have recently been proposed. The one we focus on here was described by Regot, et al. (2011), and uses engineered yeast cells as building blocks for the evaluation of logical functions. Each cell type implements a specific logic function, and, by combining cells together, the authors were able to evaluate a large number of complex circuits (including a multiplexer and an adder) using a relatively small “library”. The basic scheme is represented (through an example) in Figure 4; cells can take inputs from other cells, from external sources, or a combination of the two. The “output” of one cell type can either be the production of some “wiring” molecule, or a reporter (e.g., fluorescence).

In Figure 4 we show how two different cell types may be combined to evaluate the Boolean NOT (negated OR) function, which returns “true” (or 1) if and only if both its inputs are equal to “false” (or 0). NOR is an important function, as it

offers a *universal basis* (that is, any arbitrary Boolean function may be converted, with a necessary increase in size, to a function using only NOR), and it already serves as a test-bed for new approaches in synthetic biology (Goñi-Moreno & Amos, 2012; Tamsir, et al., 2011). One cell implements the NOT function, and the other implements the N-IMPLY function (which takes two inputs, but evaluates to “true” only if one specific input is “true”). In this case, when both C and B are “true”, then C is negated, and the output is “false”. However, if *only* B is “true”, then the “false” input to C is negated to “true”, and the whole function evaluates to “true” (obviously, the function cannot evaluate to “true” in the absence of a “true” input to B, in which case the state of C is irrelevant). By connecting both cells with a pheromone signal “wire”, Regot, et al. (2011) obtained a reliable NOR gate. They demonstrated their approach experimentally, and showed how a number of non-trivial, complex circuits could be constructed. In particular, they showed how a *multiplexer* (that is, a circuit that selects one of several inputs and “feeds” it to a specific output) may be constructed using just three engineered cell types, three distinct input signals, and a single “wiring” molecule. As the authors point out (with a certain degree of understatement), “This circuit, if designed in a single cell would be difficult to implement *in vivo*.”

A similar, distributed approach is taken by Tamsir, et al. (2011), who describe (in the same issue of *Nature*) the construction of a multi-cellular NOR gate, using quorum sensing molecules as “wires”. They demonstrate how their NOR gates may be combined in order to evaluate other functions, including the notoriously difficult XOR. In an accompanying commentary article, Li & You (2011) describe the benefits of both approaches: (1) *encapsulation*: the notion that each logic gate, when engineered into a cell, may be considered to be a “black box”, with implementation details hidden from the designer (that is, the designer need only concern themselves with module interfacing issues), (2) the facility for module *reuse* (which derives naturally from encapsulation): modules are general-purpose, and may be used multiple times, in different parts of a circuit, and (3) *noise suppression*: the use of circuit layers containing a sub-population of cells allows erroneous individual cell responses to be filtered out.

Although distributed cellular computation is still at a relatively early stage of development, researchers are already seeking to augment the range of *communication schemes* that may be employed. The two main problems with using signalling molecules or pheromones for inter-cellular communication is that (1) they generally offer a single message per channel, due to the nature of the molecular “docking” that transmits the information, and (2) the nature of the message itself is restricted to a particular functional type (e.g., “regulate transcription”) Ortiz & Endy (2012). As the authors of that paper point out, one possible approach for overcoming these limitations may be to “. . . establish an information channel that is capable of transmitting arbitrary messages encoded via a common format.” The solution they demonstrate involves the exchange of *genetic material*, rather than simple signalling molecules, as these offer the capability of transmitting arbitrary messages. Their “DNA messaging” protocol uses bacterial *transduction* - the transfer of DNA from one cell to another by a virus (in this case, M13) Russel (1995). The three success criteria defined for their system were (1) *decoupling*: the ability to transmit different messages in a “reusable” fashion (see the “reuse” benefit in the previous paragraph), (2) *flexibility*: the ability to transmit messages of different length, and messages that specify different functional outcomes, and (3) *specificity*: the ability to target messages to particular cell types in a mixed population. All three criteria were met in a series of experiments; moreover, Ortiz & Endy (2012) also demonstrated the ability to extend the range of their messaging scheme by

harnessing the power of bacterial *chemotaxis* (the ability of cells to move in response to chemical gradients). This allowed them to achieve DNA messaging at the centimetre scale.

An alternative messaging scheme was also proposed by Goñi-Moreno, et al. (2013), which uses bacterial *conjugation* (transfer of DNA during direct cell-cell contact, which has been likened to “bacterial sex”) Llosa, et al. (2002). Although this scheme has yet to be tested in the laboratory, the authors argue that it offers potential benefits in terms of *scalability*, and “conjugation computing” is the subject of an ongoing research project¹.

5. Conclusions and future directions

In this paper we have described an emerging “third wave” of synthetic biology, in which attention is moving from single-cell solutions, towards “networked”, multicellular approaches. Such models will allow us to transcend the inherent limitations of isolated cells, and enable the full potential of biological “wetware”. As Bacchus & Fussenegger (2013) argue, “The move from intracellular to intercellular communication systems is a major tool that will enable future advances in synthetic biology. These intercellular systems are likely to out-perform any intracellular counterpart. The future will require greater processing capacity, as more complex networks and circuits are being established, which will dramatically increase performance characteristics. Therefore, the division of metabolic workload of the overall system between cells is crucial, as one cell is unlikely to be sufficient. Just as computers are assembled using different standardized electrical hardware coupled in a rational way to increase a systems overall performance, the assembly of distinct biological cellular machineries will be achieved in a similar manner.” Of course, this transition is not without its difficulties, and some of the major future challenges include (1) how to ensure long-term maintenance of homeostasis in engineered consortia, (2) how to account for (or even harness) horizontal gene transfer between population members, (2) how to engineer “non-standard” organisms that offer new and useful functionalities, and (4) how to “fine tune” the behaviour of multiple interacting microbial populations Brenner, et al. (2008). Computational modelling and simulation will provide central contributions to addressing these, and the ongoing development of software tools (see Slusarczyk, et al. (2012) for a recent list) will continue to enable fruitful collaborations between biologists, mathematicians, computer scientists and engineers.

Acknowledgements

Parts of this paper were originally developed for a series of lectures I presented at the Universidad Politécnica de Madrid. I thank Alfonso Rodríguez-Patón for the invitation, and the students for their useful comments and questions.

¹See the EU-funded PLASWIRES project (2013-2016): <http://www.plaswires.eu/>

References

- Amos, M. (2006). *Genesis Machines: The New Science of Biocomputing*. London: Atlantic Books.
- Andrianantoandro, E., Basu, S., Karig, D. K., & Weiss, R. (2006). Synthetic biology: new engineering rules for an emerging discipline. *Molecular Systems Biology*, 2(1).
- Atkinson, S., & Williams, P. (2009). Quorum sensing and social networking in the microbial world. *Journal of the Royal Society Interface*, 6(40), 959-78.
- Attiya, H., & Welch, J. (2004) *Distributed Computing: Fundamentals, Simulations and Advanced Topics*. Hoboken, NJ: John Wiley and Sons.
- Bacchus, W., & Fussenegger, M. (2013). Engineering of synthetic intercellular communication systems. *Metabolic Engineering*, 16, 33-41.
- Balagaddé, F. K., Song, H., Ozaki, J., Collins, C. H., Barnet, M., Arnold, F. H., Quake, S. R., & You, L. (2008). A synthetic *Escherichia coli* predator-prey ecosystem. *Molecular Systems Biology*, 4, 187.
- Ball, P. (1999) *The Self-Made Tapestry: Pattern Formation in Nature*. Oxford: Oxford University Press.
- Basu, S., Gerchman, Y., Collins, C. H., Arnold, F. H., & Weiss, R. (2005). A synthetic multicellular system for programmed pattern formation. *Nature*, 434(7037), 1130-4.
- Beckskei, A., & Serrano, L. (2000). Engineering stability in gene networks by autoregulation. *Nature*, 405(6786), 590-593.
- Benner, S. A., & Sismour, A. M. (2005). Synthetic biology. *Nature Reviews Genetics*, 6(7), 533-43.
- Bray, D. (2009). *Wetware: A Computer in Every Living Cell*. New Haven, CT: Yale University Press.
- Brenner, K., You, L., & Arnold, F. H. (2008). Engineering microbial consortia: A new frontier in synthetic biology. *Trends in Biotechnology*, 26(9), 483-9.
- Church, G., & Regis, E. (2012). *Regenesis: How Synthetic Biology will Reinvent Nature and Ourselves*. New York, NY: Basic Books.
- Danino, T., Mondragón-Palomino, O., Tsimring, L., & Hasty, J. (2010). A synchronized quorum of genetic clocks. *Nature*, 463(7279), 326-30.
- Dorrián, H., Borresen, J., & Amos, M. (2013) Community structure and multimodal oscillations in complex networks. *PLOS ONE* 8(10): e75569.
- Elowitz, M. B., & Leibler, S. (2000). A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767), 335-338.
- Forster, A. C., & Church, G. M. (2006). Towards synthesis of a minimal cell. *Molecular Systems Biology*, 2, 45.
- Gardner, T. S., Cantor, C. R., & Collins, J. J. (2000). Construction of a genetic toggle switch in *Escherichia coli*. *Nature*, 403, 339-342.
- Goñi-Moreno, A., & Amos, M. (2012). A reconfigurable NAND/NOR genetic logic gate. *BMC Syst Biol*, 6(1), 126.
- Goñi-Moreno, A., Amos, M., & de la Cruz, F. (2013). Multicellular computing using conjugation for wiring. *PLOS ONE*, 8(6), e65986.
- Goodwin, B. C. (1965). Oscillatory behaviour in enzymatic control processes. In Weber, G. (Ed.), *Advances in Enzyme Regulation*, Vol. 3, p. 425-438. Oxford: Pergamon Press.
- Hasty, J., McMillen, D., & Collins, J. J. (2002). Engineered gene circuits. *Nature*, 420(6912), 224-30.
- Heinemann, M., & Panke, S. (2006). Synthetic biology - putting engineering into biology. *Bioinformatics*, 22(22), 2790-9.

- Khalil, A. S., & Collins, J. J. (2010). Synthetic biology: Applications come of age. *Nature Reviews Genetics*, 11(5), 367-79.
- Levy, S. (1992). *Artificial Life: The Quest for a New Creation*. London: Jonathan Cape.
- Li, B., & You, L. (2011). Division of logic labour. *Nature*, 469(4), 171-172.
- Likic, V. A., McConville, M. J., Lithgow, T. & Bacic, A. (2010). Systems biology: The next frontier for bioinformatics. *Advances in Bioinformatics*. doi: 10.1155/2010/268925.
- Llosa, M., Gomis-Rüth, F. X., Coll, M., & Cruz, F. D. L. (2002). Bacterial conjugation: a two-step mechanism for DNA transport. *Molecular Microbiology*, 45(1), 1-8.
- Lou, C., Liu, X., Ni, M., Huang, Y., Huang, Q., Huang, L., et al. (2010). Synthesizing a novel genetic sequential logic circuit: A push-on push-off switch. *Molecular Systems Biology*, 6, 350.
- Macía, J., Posas, F., & Solé, R. V. (2012). Distributed computation: The new wave of synthetic biology devices. *Trends in Biotechnology*, 30(6), 342-9.
- McDaniel, R., & Weiss, R. (2005). Advances in synthetic biology: On the path from prototypes to applications. *Current Opinion in Biotechnology*, 16(4), 476-83.
- Monod, J. (1971). *Chance and Necessity*. Alfred A. Knopf.
- Noireaux, V., & Libchaber, A. (2004). A vesicle bioreactor as a step toward an artificial cell assembly. *Proceedings of the National Academy of Sciences of the United States of America*, 101(51), 17669-17674.
- O'Malley, M. A. & Dupré, J. (2005). Fundamental issues in systems biology. *BioEssays* 27, 1270-1276.
- Ortiz, M. E., & Endy, D. (2012). Engineered cell-cell communication via DNA messaging. *Journal of Biological Engineering*, 6(1), 16.
- Payne, S., Li, B., Cao, Y., Schaeffer, S., Ryser, M. D. & You, L. (2013). Temporal control of self-organized pattern formation without morphogen gradients in bacteria. *Molecular Systems Biology*, 9:697.
- Pohorille, A., & Deamer, D. (2002). Artificial cells: Prospects for biotechnology. *Trends in Biotechnology*, 20(3), 123-128.
- Purnick, P. E., & Weiss, R. (2009). The second wave of synthetic biology: From modules to systems. *Nature Reviews Molecular Cell Biology*, 10(6), 410-22.
- Rasmussen, S., Chen, L., Nilsson, M., & Abe, S. (2003). Bridging nonliving and living matter. *Artificial Life*, 9(3), 269-316.
- Regot, S., Macia, J., Conde, N., Furukawa, K., Kjellén, J., Peeters, T., Hohmann, S., de Nadal, E., Posas, F., & Solé, R. (2011). Distributed biological computation with multicellular engineered networks. *Nature*, 469(7329), 207-11.
- Rinaudo, K., Bleris, L., Maddamsetti, R., Subramanian, S., Weiss, R., & Benenson, Y. (2007). A universal RNAi-based logic evaluator that operates in mammalian cells. *Nature Biotechnology*, 25(7), 795-801.
- Ruoff, P., Vinsjevik, M., Monnerjah, C., & Rensing, L. (1999). The Goodwin oscillator: On the importance of degradation reactions in the circadian clock. *Journal of Biological Rhythms*, 14(6), 469-479.
- Russel, M. (1995). Moving through the membrane with filamentous phages. *Trends in Microbiology*, 3(6).
- Serrano, L. (2007). Synthetic biology: Promises and challenges. *Molecular Systems Biology*, 3, 158.
- Slusarczyk, A. L., Lin, A., & Weiss, R. (2012). Foundations for the design and implementation of synthetic genetic circuits. *Nature Reviews Genetics*, 13(6), 406-20.
- Solé, R. V., & Macia, J. (2013). Expanding the landscape of biological computation

- with synthetic multicellular consortia. *Natural Computing*. doi:10.1007/s11047-013-9380-y.
- Strogatz, S. (2003). *Sync: The Emerging Science of Spontaneous Order*. London: Penguin Books.
- Szybalski, W. (1974) Contribution to *Eighteenth OHOLO Biological Conference* panel discussion. In Kohn, A., & Shatky, A., (Eds.), *Control of Gene Expression*, p. 405. New York: Plenum Press.
- Tabor, J. J., Salis, H. M., Simpson, Z. B., Chevalier, A. A., Levskaya, A., Marcotte, E. M., Voigt, C. A., & Ellington, A. D. (2009). A synthetic genetic edge detection program. *Cell*, 137(7), 1272-81.
- Tamsir, A., Tabor, J. J., & Voigt, C. A. (2011). Robust multicellular computing using genetically encoded NOR gates and chemical 'wires'. *Nature*, 469(7329), 212-5.
- von Neumann, J. (1951). The general and logical theory of automata. In Taub, A. H., (Ed.), *Collected Works of John von Neumann, Vol. 5*, p. 289-326. Wiley.
- Wall, M. E., Hlavacek, W. S., & Savageau, M. A. (2004). Design of gene circuits: Lessons from bacteria. *Nature Review Genetics*, 5(1), 34-42.