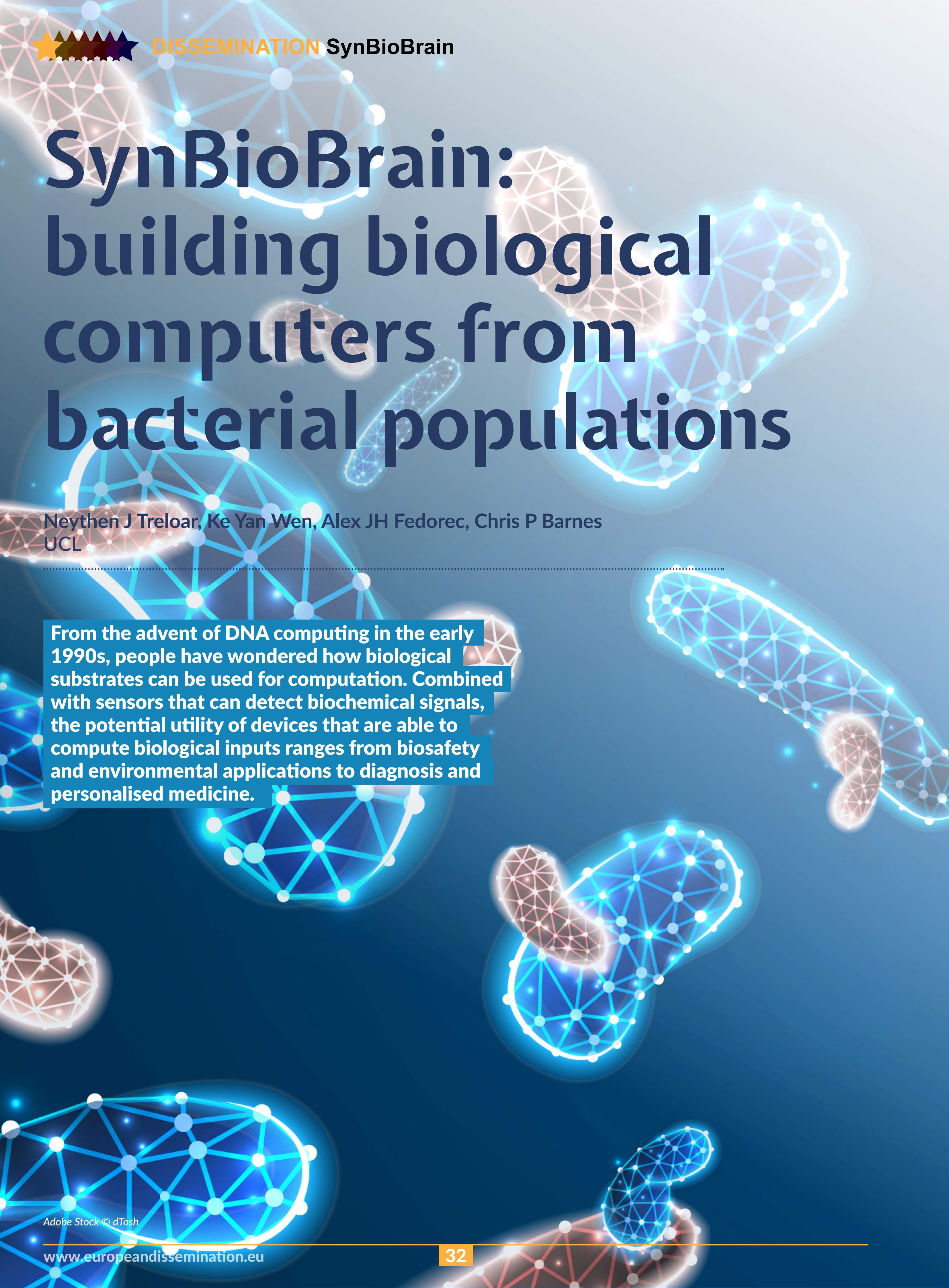


SynBioBrain: building biological computers from bacterial populations

Neythen J Treloar, Ke Yan Wen, Alex JH Fedorec, Chris P Barnes
UCL

From the advent of DNA computing in the early 1990s, people have wondered how biological substrates can be used for computation. Combined with sensors that can detect biochemical signals, the potential utility of devices that are able to compute biological inputs ranges from biosafety and environmental applications to diagnosis and personalised medicine.

Adobe Stock © dTosh

Previously there has been a major effort in the field of synthetic biology to engineer digital logic within single strains of bacterial, yeast, or mammalian cells using genetic logic gates. While this approach can implement complex truth tables, new strains have to be engineered for each application, which is costly and time-consuming. Here we are interested in engineering biology at a larger scale, using microbial consortia located across space. These offer a powerful platform in which information is integrated and processed in a distributed fashion. This approach is inspired by natural processes such as patterning in embryo development and physiological differentiation in bacterial biofilms, which utilise chemical gradients to organise behaviour. Here a computer is composed of a number of bacterial colonies which communicate using diffusible morphogen-like signals. Their behaviour and response to the signalling molecules can be monitored by coupling to the expression of fluorescent proteins. A computation is programmed into the overall physical layout of the system by arranging colonies such that the resulting diffusion field encodes the desired function, and the output is represented in the fluorescent spatial pattern displayed by the colonies.

This approach is radically different to the previous synthetic distributed computing examples because it performs spatial computation depending on the arrangement of the different colonies. This means that once biosensing strains are developed for a particular chemical or metabolite and engineered to produce an intercellular signalling molecule in response, they can be arranged spatially to perform different computations. We envisage a ‘living microchip’ formed from colonies printed on agar plates or paper substrate.

Our approach will open up new ways to perform biological computation with synthetic biology, bioengineering, and biosensing applications. Ultimately, these computational bacterial communities will help us explore information processing in natural biological systems.

> Our approach will open up new ways to perform biological computation with synthetic biology, bioengineering, and biosensing applications.

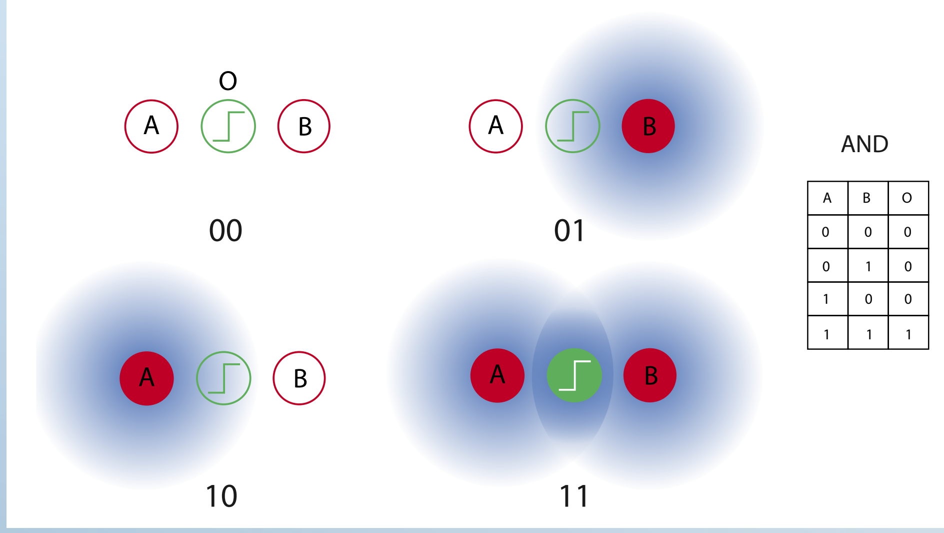
How can computation be done across space?

Imagine we have three colonies of bacteria arranged in space as in Figure 1. Two of the colonies are formed from sender bacteria which can send out a diffusible molecule when activated (denoted as A and B). The third colony is known as a receiver, and it is formed from bacteria that detect the diffusible molecule and express a green fluorescent protein (denoted as O). The response of the receiver turns on sharply in response to the signalling molecule (depicted by the step function).

Now we can have four different scenarios depending on whether the colonies A and B are ON and sending their signalling molecule. Each of the four panels describes the case with A and B OFF (00), only B ON (01), only A ON (10), and both A and B ON (11). Depending on the properties of the receiver colony and the distance of the receiver from the senders, it can be made to be ON only in the last case (11). This logic describes a very common digital computation known as an AND gate. The computation can be represented in a **truth table** where the columns denoted A and B correspond to the inputs and O represents the output colony.

The mathematics of spatial computation

To make mathematical statements about the capability of the system, we developed an abstract representation. This representation was inspired by statistical physics and allows us to enumerate the distinct logic functions possible given a certain number of sender and receiver colonies. We consider the four input states as a sequence from left to right in



A	B	O
0	0	0
0	1	0
1	0	0
1	1	1

Figure 1: Overview of spatial computing. Three colonies are placed in a row, two of which are sender colonies, and one is a receiver. When the sender colonies are ON they send out diffusible signalling molecules (depicted in the 01, 10 and 11 panels by the blue shading). By suitable choice of the response of the receiver and the spatial arrangement, only the case where both senders are activated causes the receiver to fluoresce green (11 panel). Together these cases give rise to a logic gate known as an AND gate.

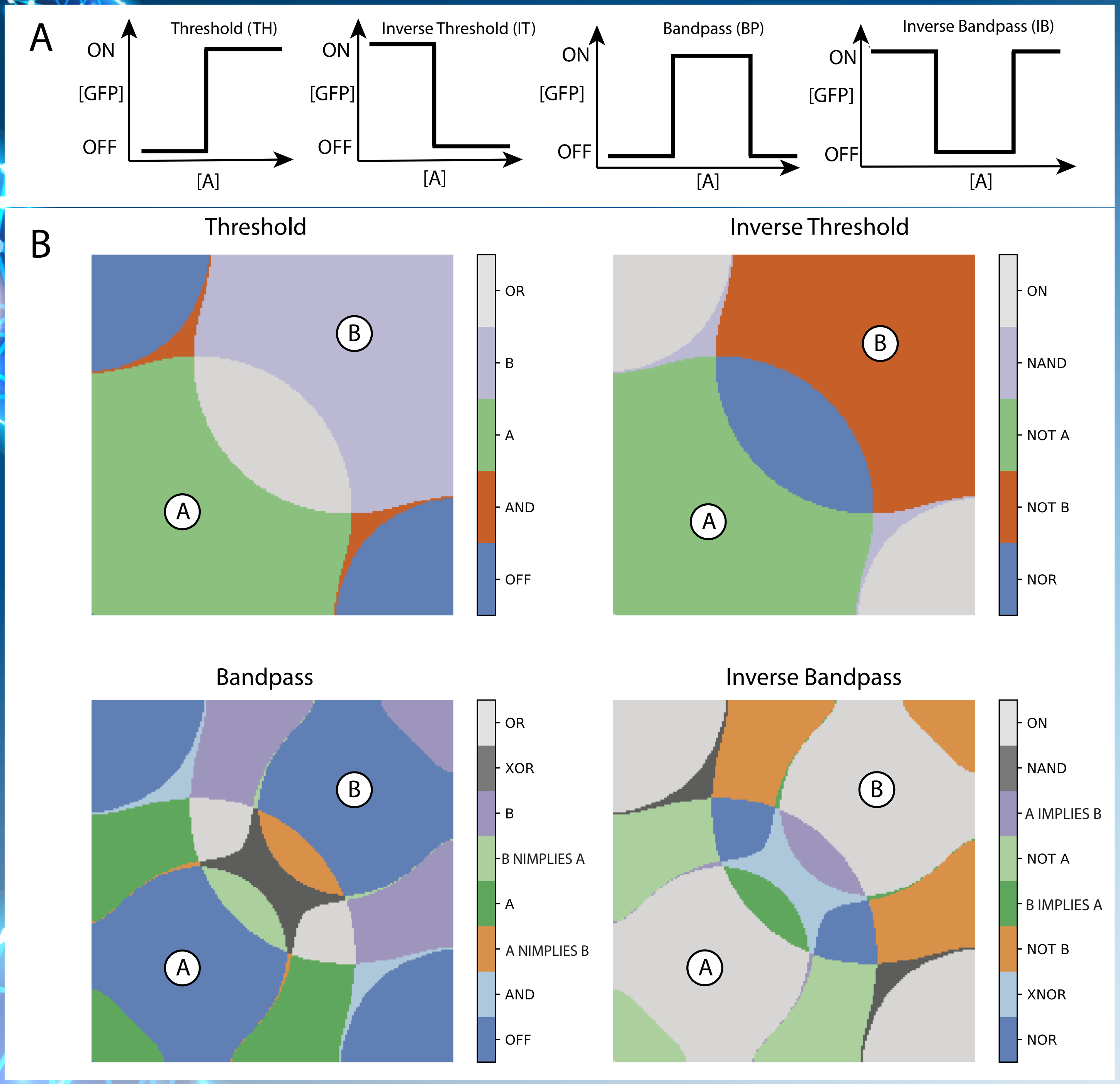


Figure 2: All sixteen two-input logic gates can be computed spatially given the four receiver response functions. A) Required receiver response functions, where [A] is the concentration of diffusible signalling molecule. B) Computational simulations verifying our mathematical approach.

A

ABC	Output
000	0
001	0
010	1
011	1
100	0
101	1
110	1
111	1

ABC	Output
000	0
001	0
010	1
011	1
100	1
101	1
110	0
111	1

ABC	Output
000	1
001	0
010	1
011	0
100	1
101	1
110	1
111	0

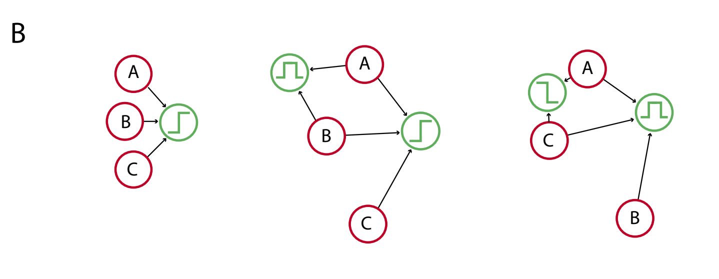


Figure 3: Arbitrary digital functions can be created by arranging senders and receivers in different configurations. A) Examples of three-input (A, B and C) truth tables representing complex digital logic. B) The implementation in the spatial biocomputer.

order of increasing signal concentration. The relative concentrations of the input states can change depending on the position of the receiver colony. For example, if the receiver is very close to input A, the order would be 00, 01, 10, 11 as the signalling molecule concentration at the receiver for 10 would be higher than 01. Similarly, if the output colony were placed close to B, the order would be 00, 10, 01, 11.

Furthermore, we represent the activation functions as boundaries partitioning the input states into ON and OFF boxes. For example, the threshold partitioning would be OFF|ON where anything to the left of the boundary is mapped to OFF, and anything to the right is mapped to ON. This represents the activation pattern of the threshold, which is inactive at low signalling molecule concentrations and active at high concentrations. Therefore, the AND gate in Figure 1 would be represented as 00, 01, 10 | 11, where the 11 input state is alone on the right of the boundary as it is the only one in which the output colony is activated.

Using this representation, we can enumerate all the distinct logic gates that are possible given the properties of the system. We used this reasoning to show that all possible two-input logic functions can be computed spatially, provided we have receivers with response functions given in Figure 2A. We verified this with computer simulations (Figure 2B). This is effectively a map of the area around the senders, where the colour of each region represents the logic gate that would be encoded if a receiver was placed in that region. We can clearly see that the function encoded by a given receiver is dependent on its position relative to the inputs and that between the four activation functions we can encode all 16 two input logic gates as predicted.

We then went on to show that we can construct any Boolean function by using two layers of digital logic (effectively multiple different receiver colonies). In this case, the function output is ON if any receiver colony fluoresces. This demonstrates the strength of the approach and proves that arbitrary spatial computation is possible. A key consideration when building electronic digital circuits is electronic design automation (EDA). EDA comprises a category of algorithms that find the simplest implementation of a given function using the available electronic parts. We have developed an analogous design algorithm that given a specified digital function, finds the simplest spatial pattern that implements the function using the smallest number of bacterial colonies. Figure 3 shows simplified designs for three different Boolean functions.

Future directions

- We are currently engineering *E. coli* strains that function as senders and receivers. We will verify our mathematical approach experimentally and generate configurations that can compute arbitrary digital functions.
- The approach to building arbitrary analogue functions will be extended.
- We will investigate more complex computing based on cellular automata (CA). These are discrete dynamical systems comprising 'cells' on a grid that update their state based on the states of their neighbours. These simple local rules are well known to generate complex behaviours but can also perform computation through changing spatial patterns and can be equivalent to Turing machines and neural networks. This will allow different computations, examples of which are temporal logic (which signals arrived first), counting of events, and supervised learning (distinguishing between sets of inputs). This architecture will enable the next generation of computational biosensor devices.

PROJECT NAME

SynBioBrain: Building biological computers from bacterial populations

PROJECT SUMMARY

Using synthetic biology, we can now engineer bacteria into whole-cell biosensors where sensing, transduction and output occur within the living cell. Applications include the detection of harmful environmental agents, bioprocess monitoring, and detecting medically relevant biomarkers. Engineering microbial consortia allows biosensor information to be integrated and processed in a distributed fashion. In this project, we construct biological computers formed from engineered bacterial populations that communicate using intercellular signalling molecules. Our approach opens up new ways to perform biological computation.

PROJECT LEAD

Professor Barnes originally received a PhD in particle physics then became interested in collective and emergent behaviour in biological systems. He joined UCL's Faculty of Life Sciences in 2012. Chris's research is original and ground-breaking, and he has built an outstanding funding and publication record. His research is multidisciplinary, combining mathematical modelling, Bayesian statistical inference, machine learning and molecular biology.

CONTACT DETAILS

Professor Chris Barnes
519 Rockefeller Building
21 University Street
London WC1E 6BT

christopher.barnes@ucl.ac.uk

https://ucl-cssb.github.io/

@cssb_lab

FUNDING

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No. 770835.