Theory Biosci.
DOI 10.1007/s12064-010-0118-4

REVIEW

Computing with bacterial constituents, cells and populations: from bioputing to bactoputing

Vic Norris · Abdallah Zemirline · Patrick Amar · Jean Nicolas Audinot · Pascal Ballet · Eshel Ben-Jacob · Gilles Bernot · Guillaume Beslon · Armelle Cabin · Eric Fanchon · Jean-Louis Giavitto · Nicolas Glade · Patrick Greussay · Yohann Grondin · James A. Foster · Guillaume Hutzler · Jürgen Jost · Francois Kepes · Olivier Michel · Franck Molina · Jacqueline Signorini · Pasquale Stano · Alain R. Thierry

Received: 5 January 2010 / Accepted: 15 December 2010

Abstract The relevance of biological materials and processes to computing—*alias bioputing*—has been explored for decades. These materials include DNA, RNA and proteins, while the processes include transcription, translation, signal transduction and regulation. Recently, the use of bacteria themselves as living computers has been explored but this use generally falls within the classical paradigm of computing. Computer scientists, however, have a variety of problems to which they seek solutions, while microbiologists are having new insights into the problems bacteria are solving and how they are solving them. Here, we envisage that bacteria might be used for new sorts of computing. These could be based on the capacity of bacteria to grow, move and adapt to a myriad different fickle environments both as individuals and as populations of bacteria plus bacteriophage. New principles might be based on the way that bacteria explore phenotype space via hyperstructure dynamics and the fundamental nature of the cell cycle. This computing might even extend to developing a high level language appropriate to using populations of bacteria and bacteriophage. Here, we offer a speculative tour of what we term *bactoputing*, namely the use of the natural behaviour of bacteria for calculating.

V. Norris · A. Zemirline · P. Amar · P. Ballet · G. Bernot · G. Beslon · A. Cabin · E. Fanchon · J.-L. Giavitto · N. Glade · P. Greussay · Y. Grondin · G. Hutzler · F. Kepes · O. Michel · F. Molina · J. Signorini · A. R. Thierry
Epigenomics Project, Genopole Campus 1, Bât. Genavenir 6, 5 rue Henri Desbruères, 91030 Évry Cedex, France

V. Norris (✉) · A. Cabin · Y. Grondin
Assemblages Moleculaires: Modelisation et Imagerie SIMS, EA 3829, Faculty of Science, University of Rouen, 76821 Mont-Saint-Aignan, France
e-mail: victor.norris@univ-rouen.fr

A. Zemirline · P. Ballet
LISyC, EA3883, UBO, 20 avenue Le Gorgeu C.S. 93837 BP 809, 29238 Brest Cedex 3, France

P. Amar
Laboratoire de Recherches en Informatique, University of Paris 11 & CNRS UMR 8623, 15 avenue George Clémenceau, 91405 Orsay Cedex, France

J. N. Audinot
Département Science et Analyse des Matériaux, Centre de Recherche Public-Gabriel Lippmann, 4422 Belvaux, Luxembourg

E. Ben-Jacob
School of Physics and Astronomy, Tel Aviv University, 69978 Tel Aviv, Israel

G. Bernot
Laboratoire I3S, Algorithmes-Euclide-B, University of Nice-Sophia Antipolis, 06903 Sophia Antipolis Cedex, France

G. Beslon
Computer Sciences Department, Institut National des Sciences Appliquées de Lyon, Bâtiment Blaise Pascal, 69621 Villeurbanne Cedex, France

E. Fanchon · N. Glade
Laboratoire TIMC-IMAG, Faculty of Médecine, 38706 La Tronche, France

J.-L. Giavitto · G. Hutzler
Laboratory IBISC-LIS project, CNRS FRE 2873, University of Evry-Val d'Essonne, France Université de Paris XII, Évry, France

P. Greussay · J. Signorini
Advanced Computing Laboratory of Saint-Denis, University Paris 8, 93526 Saint-Denis, France

Published online: 08 March 2011

Keywords Biological computing · Bacteria · Minimal cell · Synthetic biology · Turing · Origin of life · Computer science

Introduction

If the two species, microbiologists and computer scientists, are to interact fruitfully, microbiologists need to have an idea of some of the problems that are of interest to computer scientists whilst computer scientists need to see solutions—perhaps to other problems—in the knowledge and intuitions of microbiologists.

What is computing? Defined narrowly, it is the systematic study of algorithmic processes that describe and transform information: their theory, analysis, design, efficiency, implementation, and application. The fundamental question underlying all computing is 'What can be (efficiently) automated?' (Denning et al. 1989). In essence, a Turing machine is a very simple computer. The Turing machine is further specified by a set of instructions which we can think of as a programme. What can a Turing machine do or not do? To answer this, consider a *Universal Turing machine* which is a Turing machine able to read the description of any other Turing machine and to do what that other Turing machine can do. A Universal Turing machine can therefore perform any definite method and, importantly, it could do this without being extraordinarily complex provided it has an immense storage capacity. (Note that a modern computer runs a *microprogramme* that allows its *processor chip* to take instructions from the main store and compute local functions of them so as to make these instructions resemble those of a particular processor; hence by changing the microprogramme, the computer becomes a PC or a Mac, or any other known computer. Most modern computers are therefore Universal Turing machines). No-one has yet found a plausible model of computation which is *more* powerful than the Turing machine. Whether living systems constitute—or could be turned into—more powerful calculating devices than Turing machines is highly controversial (see for example Ben-Jacob 2003; Ben-Jacob and Shapira 2004). It has been suggested, for example, that when the green sulphur bacterium, *Chorobium tepidum*, transfers and traps light energy, it actually performs a quantum computation in using a wavelike characteristic of the energy transfer within the photosynthetic complex to allow the complexes to sample simultaneously different states and find the most efficient path (Engel et al. 2007); this can be likened to an algorithm in quantum computing for searching an unsorted information database (Grover 1997). That said, as coauthors with differing opinions, we choose in the following to skirt the issue of whether cells offer an alternative to the paradigm of the Turing machine.

What is a cell? It can be argued that the cell is an autocatalytic network (Kauffman 1996), or a neural net, or a tensegrity structure (Ingber 1998), or a pattern of connectivity with characteristics of Small Worlds and Self-Organised Criticality (Barabasi and Oltvai 2004), or a giant oscillating dipole (Fröhlich 1978), or a unit of subjective experience (Norris 1998) etc. It seems evident that the cell is the creator and the creation of an extraordinarily high density of different organising processes that have autocatalytic relationships with one another (Norris et al. 2004). It is a system that produces self-organisation and assembly by recruiting and dismissing a multitude of processes and molecules. An exciting question for bioputer designers is therefore what else does this? What else, in other words, could be modelled using a cell and, in particular, a bacterial cell?

What is bioputing? The relevance of certain biological materials and processes to computing has been understood for decades. These materials include DNA, where its value to different sorts of computing, such as the solution of combinatorial problems, is well-known (Adleman 1994; Carbone and Seeman 2002; Rothemund et al. 2004). Such materials in combination with biological processes can constitute effective computers (Benenson et al. 2004). Hence, bacteria and other cells can be used as a source of new materials with new properties for computing along traditional lines. They may also be used in an intact form for simple forms of such computing (Skretas and Wood 2005; Basu et al. 2005). Attempts to construct the minimal cell, inspired in part by origins of life studies and by biotechnological applications, may also produce cells that are amenable for sophisticated, albeit traditional, computing. All these approaches form part of the general approach of

J. A. Foster
Department of Biological Sciences, University of Idaho, Moscow, ID 83844-3051, USA

O. Michel
Lab. LACL-Bât P2–240, Faculté des sciences et technologies, 61 Av. du Général de Gaulle, 94010 Créteil Cedex, France

F. Molina · A. R. Thierry
Sysdiag UMR 3145-CNRS/BIO-RAD, Cap Delta 1682 Rue de la Valsiere, 34184 Montpellier Cedex 4, France

P. Stano
Enrico Fermi Research Centre, Compendio del Viminale, 00184 Rome, Italy

J. Jost
Max Planck Institute for Mathematics in the Sciences, Inselstr.22, 04103 Leipzig, Germany

J. Jost
Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501, USA

what we term *bioputing*. At this early stage of the game, it may be sensible to make the definition as broad and flexible as possible, and we therefore consider that bioputing is the use for computing of biological materials or biological structures or indeed biology-inspired functions.

What is *bactoputing*? Bactoputing is the use for computing of actual bacteria or of functions inspired by bacteria. Bactoputing therefore includes the use of the natural behaviour of bacteria for computing and, as such, it is a subset of bioputing. The natural behaviour of bacteria includes chemotaxis and sporulation. In chemotaxis towards an attractant, the swimming bacterium measures the nutrient level, stores this information and then processes this information by comparing it with a new measurement; if the level has increased, the bacterium is likely to continue running in a straight line, while if the level has not increased it is likely to tumble and then run in a new direction. In sporulation, individual cells assess the level of nutrients and the level of stress (their own and that of the other cells) and vote for or against sporulation in what is both an individual and a collective decision. Here, we try to focus on a version of bactoputing in which bacteria are considered as computers. One common approach to computing with bacteria entails adapting them, so that they become identical sets of logic gates. Each essentially identical bacterium is then a constituent of a computer; in other words, a homogeneous population of bacteria constitutes the computer. In this approach, one possibility is to use the logic systems that are native to the bacterium, to use, in other words, its original set of networks of gene expression and protein synthesis (Thomas 1980; Gardner et al. 2000; Atkinson et al. 2003; Ozbudak et al. 2004; Wall et al. 2004). One of the obvious attractions here would be the capacity of bacteria to multiply cheaply. One of the drawbacks is that bacteria have a tendency to follow their own agenda and frustrate attempts to engineer them to follow human designs (but see below Posfai et al. 2006). An alternative approach is to consider each bacterium as different (Tolker-Nielsen et al. 1998; Booth 2002; Balaban et al. 2004; Avery 2006). In this case, a heterogeneous population of bacteria constitutes the bactoputer. This is the tack we follow here. We choose to ignore a different version of bactoputing in which bacteria are considered as agents that both compute and act; for example, bacteria may be modified to recognise, invade and treat cancer cells or parasites within us (Baker 2005). This would involve adapting what certain species of bacteria do anyway, and is therefore in the spirit of bactoputing. More speculatively still, bacteria might be converted into new organelles in a remake of the origins of eukaryotic cells. Such organelles might function to repair the host cell and reverse ageing (Norris, in preparation). This would entail making full use of the capacity of bacteria to sense their environment and to modify it. But again we choose to ignore this in what follows.

In section "A few questions in computer science and the social sciences", we mention a number of problems that may, one day, be amenable to bactoputing. These problems include: many combinatorial problems that are unsolvable by traditional computing, since they entail exponential increases in the number of steps needed; hardware problems due to lack of memory or the difficulty of construction in 3D; problems faced by many social groups in which a compromise must be found so as to survive in difficult conditions but to proliferate in favourable conditions; 'undecidable' problems that may require construction of a bactoputer or other novel brain; the problem of finding a new high level language appropriate for a bactoputer. In section "Applying bactoputing to problems", we try to address the problems raised in the second section. In section "Specific experiments", we suggest specific initial experiments that might be performed. In the Appendix, we review, for computer scientists interested in bactoputing, certain aspects of microbial physiology along with efforts to construct simplified bacteria by deleting 'superfluous' DNA from genomes, by use of wall-less variants (L-forms), by selecting bacteria via mutation and selection, and by origin-of-life experiments to make liposome-based systems.

A few questions in computer science and the social sciences

NP problems

NP-hard problems would be good test cases for bactoputing because they would be a definitive demonstration of how bactoputing can speed things up (even though the real value of bactoputing may lie elsewhere, see below). But what are NP problems? If the number of steps in the calculation is given by a function of N and each step takes a microsecond, for $N = 100$, functions such as $\log_{10}N$ and N^5 are tractable since they take 2 μs and 3 h, respectively, while N^N is not tractable (it would take 3×10^{186} years). This leads to the idea that an algorithm can be tractable, if its behaviour depends polynomially (N^2, N^3, etc.) on the size, N, of input. This idea can be extended to the problem treated which is considered tractable if its worst case can be solved by a tractable algorithm. The class **P** is the class of tractable decision problems. The class **P** is about polynomial *time* but there is a wider class of problems, **PSPACE**, that are solvable with a polynomial amount of *memory*. This has a direct relevance to a bacterial population which, in the right conditions, can undergo an exponential increase in mass.

The Hamiltonian circuit problem is whether there is a route which visits every village exactly once and which ends at the village where it started. The related Travelling Salesman problem (see below) is to find the shortest route which visits every village at least once and which ends at the start. These are examples of the **NP** class of problems, which *may* be tractable but for which no polynomial-time algorithm is known. A decision problem (one that needs a 'yes' or a 'no' as an answer) is said to be in **NP** if there exists the equivalent of a lucky guess algorithm (a non-deterministic or pseudo-algorithm) for instances of the problem needing a 'yes' that takes at most polynomial time to correctly answer 'yes'. The problem of whether or not **P** and **NP** are the same class of problems is a major question in mathematics and has economic repercussions. If they are the same class and a problem in **NP** is tractable without, as well as with, a lucky guess, then much larger instances of them can be tackled. If they are different classes and problems in **NP** can be shown to be intractable, the search for certain types of algorithm for them can cease.

Problems can sometimes be transformed into one another (this is the case for the Travelling Salesman and the Hamiltonian Circuit) and, since any polynomial function of a polynomial is itself also a polynomial, as long as the time taken to do the transformation is polynomial in N, the size of the input, the time taken by a polynomial-time algorithm for the transformed problem must also be polynomial. This notion of transformation is important because many **NP** problems can transformed into a problem that is itself in **NP**; the hard problems of **NP** are termed **NP**-complete problems (an **NP**-complete problem is in **NP** and every problem of **NP** can be reduced to it by means of a polynomial transformation) (Garey and Johnson 1979). This is a general term for a wide variety of many problems, indeed there is a large array of **NP** problems involving networking, timetabling, packing, matrices, geometry, and combinatorial mathematics (note that DNA sequence comparisons are in **NP** if mismatches and gaps are allowed).

The problem of density

A recurrent problem is that computers have insufficient memory or run too slowly. One limitation to the speed at which computers can run is the distance between components. This limitation is due to what is essentially a 2D construction of the integrated circuit. The possibility of constructing a nanoscale 3D calculating device would therefore be very attractive.

Optimisation and constraint problems in organisations

Many social and economic problems require an organisation to steer between survival and growth. Companies and universities must survive hard (financial) times and expand in good ones. These appear to be contradictory constraints. No single optimal solution exists. For example, there may be no individual solution to the management problem of what proportion of the staff of a multinational group or of a research organisation should be permanent. One possibility is to consider the organisation as a collection of relatively independent units, such as research laboratories, that could offer a simultaneous diversity of independent solutions. A range of different solutions may be needed—but what is this range? Other questions where we might look to bacteria for answers include the optimum number of decision-making levels and identification of the subsystem that actually makes the decision. Finally, many social organisations are constrained by the need to reconcile coherence with their present environment and coherence with their past environments. Research laboratories have to respond to new discoveries and to new funding initiatives, but must reconcile these with their research history and, in particular with their skills, experience and interests. Perhaps bacteria have something to teach us here too.

Recognition and other problems

Electronic circuitries or even neuronal brains may be used to address complex problems that include many undecidable problems such as the recognition of shapes (e.g. is this a picture of a horse?) and optimisation problems with non-separable objective functions (e.g. the problem of attributing local rules to components so as to obtain a given global behaviour). A potential ability to address such problems is one motivation for research into the design of synthetic 'brains'. Some of these brains assemble readily into structures, are easy to understand and straightforward to control, which facilitates interfacing with users. They include self-organised networks of real neurones connected to electronic chips (Demarse et al. 2001). There are also 'soft' networks where the circuits are not fixed and easy to reconfigure. These include bioelectronic hybrid architectures such as those based on dynamic circuits made of the slime mould *Physarum polycephalum* (Tsuda et al. 2006) or 'chemical brains' based on collisions between chemical waves in the Belousov-Zhabotinsky reaction (Adamatzky and de Lacy Costello 2002). What are the possibilities for a bacterial brain?

Beyond high level instructions

High level programming languages are written in terms of instructions that include loops (for, while, do while), tests (if, switch), and operate on variables and functions (procedures and function calls). In general, each instruction is specific and instructions are acted on sequentially. How

might very different languages be developed? Biological systems have inspired imitation in conventional computing in the case, for example, of genetic algorithms. Might bacteria inspire an effectively different style of computing? Would it be possible, for example, to devise a new type of programming language based on bacterial actions?

In this context, the problem of emergence in complex systems, which Von Neumann computing cannot solve, is one that might be taken on by bactoputing (Zemirline and Norris 2007). The emergent properties of the system are those that cannot be readily predicted from a knowledge of the constituents of the system (Van Regenmortel 2004). Suppose that some emergent property P, in some complex system S, can be modelled by a function $f(x_1, x_2,\ldots,x_n)$, where for $j = 1,\ldots, n$, the variables x_j characterise, respectively, the subsystems S_j constituting S. Note that there are no functions $f_j(x_j)$ representing properties P_j of the subsystems S_j such that $f(x_1,\ldots, x_n) = f_1(x_1)$ **o**…**o** $f_n(x_n)$, where the operator **o** is an internal composition law, like +, *, /, etc. Since the emergent function $f(x_1, x_2,\ldots, x_n)$ cannot be obtained from the functions $f_j(x_j)$ (otherwise $f(x_1, x_2,\ldots, x_n)$ would be not emergent) then $f(x_1, x_2,\ldots, x_n)$ is not equal to $f_1(x_1)$ **o**…**o** $f_n(x_n)$, (thus f is said to be non-separable), and $f(x_1, x_2,\ldots, x_n)$ cannot generally be computed by silicon-based computers. Given the importance of emergence in biological systems (Van Regenmortel 2004), this may be a promising area to explore with bactoputing.

Applying bactoputing to problems

Solving the travelling salesman problem?

Quorum-sensing can be used as the basis of a population-based computing (Bulter et al. 2004; You et al. 2004). Suppose that short peptides A and B exported from two different bacteria into the medium bind to receptors in a third bacterium to initiate signal transduction (via for example well-known sensor kinases/response regulators) in this bacterium that then activates or represses synthesis of another peptide C that is exported. In principle, one could have a limitless supply of logic gates of every conceivable type. Each bacterium becomes a swimming logic gate communicating via diffusible peptides. Bacteria that have not taken part in signalling could be eliminated (for example, the sensor kinase could also induce synthesis of a factor that protects the bacterium from an externally added or an internally produced poison). This would be the equivalent of apoptosis in the brain. The numbers of an individual species of bacterium in the population become the equivalent of weights in a neural net. Proteases added to the media could be used to remove signalling peptides and so synchronise the system. In this approach, chemotaxis, which *E. coli* uses to swim up gradients of attractants (or down those of repellents), could be used to produce a structuring of the volume in the flask such that those bacteria that are attracted to others aggregate; such structuring could result in a rapid transfer of peptides between different bacteria and could be detected if different species of bacteria were to emit light of different frequencies. Refinements that might be possible include the use of a particular peptide to activate transport systems, so that whole families of gates could be switched on or off so as to construct hierarchies of gating systems. A connection with the environment could be ensured by restricting one class of peptides to be environmental signals and a second class to be the responses at the end of the line (and which could bind to biochips to trigger electrical changes). In the ideal world, learning would occur if the correct combination of response peptides were rewarded by an influx of glucose into the system.

To illustrate how peptide signalling plus differential growth might work, consider the problem of the travelling salesman who has to find the shortest route between the cities A, B, C, D, and E which he must only visit once (see above). Suppose that the shortest route is A → B→C → D→E → A and that each city is represented by a peptide. To set the problem up, we construct a bacterium that has input A and output B (denoted by A → B), another that has A → C…D → A, E → A, etc. Suppose each bacterium can only grow if it receives A, B, C, D, and E (each of which induces the expression of a different gene encoding a labile protein essential for growth) and the culture is fed in a chemostat with a limiting concentration of A, B, C, D, and E (plus everything else in excess needed for growth). This selects for an autocatalytic network based on signalling of the style A → C, C → E, E → B, B → D, and D → A. In addition, each bacterium is engineered at the start so as to produce its output in inverse proportion to the distance between the cities; hence, the greater the distance between the cities, the less the bacterium produces of the output peptide for a constant input peptide (there is the equivalent time delay possibility). Given that there is a front member and a back member in each pair of cities (A and B, respectively, in the A → B pair), we term this selection for the back member *backend* selection. The initial population must contain representatives of every pair of cities in both directions (e.g. A → B and B → A). The object is to obtain the most efficient autocatalytic network, since this should correspond to the shortest route between the cities. A problem arises because an autocatalytic network can be obtained that does not correspond to an unbroken circuit; for example, the autocatalytic network (A → C, C → E, E → B, B → D, C → A) generates peptides A, B, C, D, and E in quantities that could compete with those generated

by the ideal solution network (A → B → C → D → E → A). It is conceivable that this and other problems may be avoided or minimised by developing a *frontend* selection that requires the A part of AB, etc., so as to ensure that each receptor (A of AB or AC or AD etc.; B of BC, BD etc. is present). One way to achieve this might be to use beads sufficiently large for each to be covered with antibodies to all the receptors (i.e., anti-A, anti-B, anti-C antibodies, etc.); a small number of the antibody-coated beads/antibodies (small with respect to the number of bacteria, so that all antibodies actually have receptors bound to them) would be added to the molten agar and the beads plus attached bacteria then used to reinitiate growth (note that in addition to selecting a subset of the bacterial population that contains the entire set of receptors, this strategy would have the advantage of concentrating this subset in the agar around the bead). An alternative solution would be to have different initial populations each containing all peptide combinations but only in one direction; unfortunately, the number of such initial populations would have to be 2 raised to the number of cities.

Another problem arises because bacteria that contribute nothing to the network can still benefit from it. One solution would be to require prolonged cycles of growth in a soft gel in which the peptides can diffuse some distance from the bacteria that produce them (so that when the members of an autocatalytic network are near one another, they benefit rather than the entire population); the temperature is then raised so that the gel becomes a sol, the bacteria can be mixed, (perhaps a proportion removed to sample), and the temperature lowered to create a gel again (Fig. 1). In the ideal world of tractable, docile, well-behaved bacteria, this should result in the selection of efficient networks corresponding to solutions to good routes to the cities (for the experiments needed, see "Specific experiments"). In the early stages of selection, the size of the colony is the measure of the quality of the corresponding solution with the biggest colony corresponding to the best solution but, as the number of cycles increases, frontend selection for all receptors plus backend selection for autocatalytic growth should yield larger and larger colonies which, in the limit, would correspond to the optimal network solution. Hence, even if the autocatalytic network (A → C, C → E, E → B, B → D, D → A) gives rise to a colony, it will only be a small one.

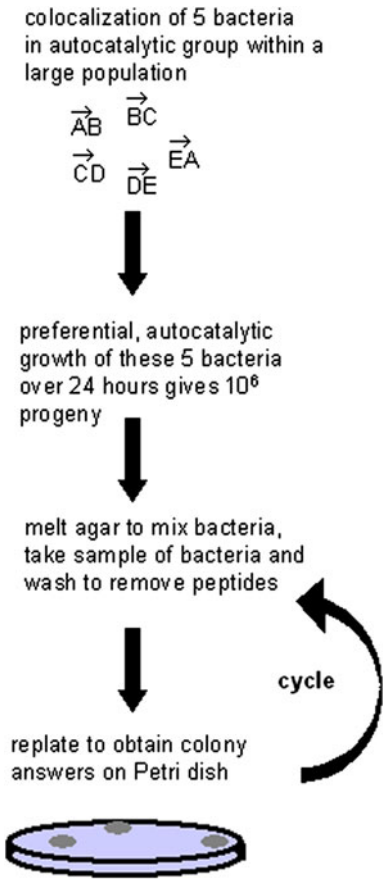

Fig. 1 Using bacteria to solve the Travelling Salesman Problem. A, B, C, D and E are five cities and the shortest path between them is in the order ABCDEA. AB represents a bacterium that on receiving peptide A exports peptide B, etc. A concentration of peptides A, B, C, D and E above a threshold is needed for rapid growth. A population containing all combinations of AB etc. is used to start the experiment but here, for simplicity, only five bacteria corresponding to the optimal combination are shown. See text for explanation ("Solving the travelling salesman problem?" and "Travelling salesman" sections)

Tackling the density problem?

Bacterial colonies are complex 3D structures in which the bacteria are densely packed (see above). A cubic millimetre of bacteria would contain at least 10^9 individual cells. In such a cube, each bacterium could act as both an element of a population-based processor and memory. Bacteria such as *E. coli* can double every 17 min so increases in computing capacity are not difficult to envisage. Moreover, in the case of a bacterial colony, the colony constructs itself as it grows (see above). The capacity of bacteria to diversify their phenotypes might also be exploited by creating conditions that allow the growth of those bacteria that take part in the calculation at the expense of those that do not take part.

Optimisation and constraint problems in organisations

Many bacterial species are extremely good at steering between survival and growth and in generating a huge diversity of behaviours in the individual bacteria that constitute a population. In trying to interpret this phenotypic diversity, it would extremely useful to obtain

information on the activity of transcription factors at the level of the individual cells within these populations rather than at the level of the average cell (which may not even exist). Given that the process of chromosome replication itself is a possible source of diversity (Rocha et al. 2003; Norris et al. 2007), it would also be useful to manipulate rates of replication (Janniere et al. 2007). Bacterial populations also generate genetic diversity, and there should be a way to make use of the mutation strategy adopted by bacteria in conditions of stress or high population densities (see above Matic et al. 2004; Kolodkin-Gal et al. 2007).

It can be argued that the phenotypes of bacteria are determined at the level of hyperstructures (see above) rather than at the level of individual macromolecules (such as genes or proteins or small signalling molecules). In this hypothesis, the bacterial population generates a range of phenotypes by varying the proportion of equilibrium and non-equilibrium hyperstructures present in each bacterium. The function that describes this variation in the population changes with different conditions and different species. How might information about this function be obtained? Ongoing developments in optical and analytic microscopy are making easier to determine which molecules and macromolecules constitute hyperstructures as well as the number and distribution of a given type of hyperstructure within a bacterial population. These techniques include *Secondary Ion Mass Spectrometry* (SIMS) which has as its main characteristics: (1) a capacity to distinguish between isotopes (such as ^{12}C and ^{13}C or ^{14}N and ^{15}N), (2) an imaging capacity with high spatial resolution (50–100 nm), and (3) a tremendous sensitivity (detection of a single protein). SIMS holds out the promise of colocalizing macromolecules in sections of cells without modification of these macromolecules (Guerquin-Kern et al. 2005; Lechene et al. 2006; Legent et al. 2008). Hence the eventual problem for students of bactoputing will be to somehow code this information into a useable form.

Recognition and other problems

It is well-known that bacteria communicate within colonies (see above). This communication is usually assumed to be chemical in nature but other possibilities should be considered, including sound (Matsuhashi et al. 1998). Chemical communication occurs by diffusion through the medium and the information may be destined for distant bacteria or for the whole population (in which case, modulation of intensity—and perhaps frequency—is important). Communication may also be strictly local and depend for example on exchange between neighbours via conjugation *pili* through which DNA can be sent.

Populations of bacteria in the form of colonies behave like huge and massively interconnected networks with seemingly intelligent behaviours. As living processors they adapt, evolve and organise themselves to process efficiently their environment and, for example, extract nutrients to transform into biomass or decide that an enemy is present. The colony can also spatially reorganise under the action of orienting perturbations such as sources of chemoattractive molecules. Adaptable, reconfigurable bacterial populations that can switch between different organisational modes (i.e. from single motile cells to colonies with well-structured morphologies, Ben-Jacob and Shapira 2004) do indeed possess the properties and capabilities needed for being chemical-biological brains. This switch corresponds to the dynamic transition of the processing system from being very efficient and globally interconnected, but poorly programmable, to being structurally programmable with a better interfacing capacity (Conrad 1995; Pfaffmann and Conrad 2000). As in a self-adaptative loop inducing the structuring and the processing ability of a network of fibrillar agents (e.g. microtubules in (Pfaffmann and Conrad 2000), the spatially distributed population of bacteria can reorganise according to the environment (e.g. it can respond to the addition at a specific time and place of chemoattractants or nutrients by reorganising spatially and functionally).

How might the problem-solving prowess of colony-forming bacteria such as *P. dendritiformis* be used in bactoputing? The use of such populations for bactoputing can be envisaged through (i) a strong interface by using a restricted list of instructions (e.g. chemical instructions, temperature or electrical stimuli,…) in order to induce specific behaviour in the bactoputer, or (ii) soft and poorly defined interfaces by the direct contact of the bactoputer with the problem it has to solve. The former category of use of a population, (i), might correspond to a transparent chip into which grooves were cut (with for example a focused ion beam) that would have diameters similar to those of a bacterium; bacteria containing fluorescent labels (fused for example to transcription factors) could then explore a network of grooves as guided by a variety of chemoattractants and chemorepellents; the grooves themselves might be curved to match the structures naturally generated within colonies. A stack of such chips, with channels connecting them, might then be made into a bacterial brain or *bactoputer* in which different distributions of types of grooves and channels would be occupied to different extents by bacteria in different states (as revealed by their fluorescence-tagged transcription factors). In the case of a chip containing several bacterial species, interspecies communication based on transfer of molecules and macromolecules might contribute to the spatial pattern within the chip and the persistence and adaptation of this pattern to differing patterns of inputs. Conceivably, such transfer might be revealed by combining SIMS and

differential labelling of different bacterial species with stable isotopes (Guerquin-Kern et al. 2005; Lechene et al. 2006; Legent et al. 2008). The latter category of use of a population, (ii), might correspond to a diagnostic chip dedicated to detection of diseases with the bacterial chip acting simultaneously as a sensor, a processor analysing complex data, and an output device that it translates this information into a form intelligible to humans. Moreover its controllability could be reinforced by driving the structuring of the colony into specific, static geometries that have been engineered (e.g. network of microscopic channels, interconnected containers…). The result would be the accomplishment of a task or macroscopically observed manifestations of the behaviour of the colonies such as the appearance of fluorescent signals or of a colony with a particular morphology.

Beyond high level instructions

If one were to devise a new programming language based on bacteria, which instructions would it contain? Some of these instructions are easy to suggest: transport (a hundred different ions and molecules); move (up and down gradients in 3D); recombine instructions (between regions of the chromosome or by making use of plasmids and transposons); exchange instructions (by conjugating or by taking up those phage that contain some chromosomal DNA); mutate (or mutate at high frequency in the case of mutator bacteria); send messages (in the form of quorum-sensing molecules and other molecules); replicate the chromosome (and pause during replication), differentiate (perhaps as a function of a role in a colony) and sporulate (or at least form a bacterium that has increased resistance); grow (at different rates); divide (to make progeny that are smaller and that may differ from one another); lyse to release phage. Other actions are harder to exploit due to the limited state of current knowledge; these include creating a hyperstructure, maintaining or altering the ratio of equilibrium to non-equilibrium hyperstructures, and increasing the diversity of hyperstructures.

How might such instructions be given? The hundreds of factors that control transcription and translation and that mediate the above actions might be manipulated chemically by fusing the genes that control them to inducible promoters (such as the one that controls the lactose operon and that can be induced by the chemical IPTG). They might also be manipulated physically by changing the temperature or exposing the bacteria to radiation or other stresses. Rather than discrete instructions being given, the instruction to the bacterial population would be in the form of a chemical or physical gradient. Hence, the instruction would be different at different places.

How would instructions be ordered? Rather than instructions simply being given in a sequence, many instructions would be given simultaneously. A metaphor for the instructions would be that of a landscape with a varied topology, different vegetation, watercourses, soil types, etc. Of course, instructions could also be given sequentially (e.g. heat shock followed by cold shock) or the possibilities inherent in pausing DNA replication might be exploited by inserting sequences into the chromosome to allow proteins to bind to them, and hence slow down or block replication in chosen regions (Laub et al. 2000; Possoz et al. 2006).

How would results be read out? When the population is a colony, results could be in the form of spatiotemporal patterns (see above) or in the distribution of extracellular signalling molecules. When the population is a suspension of cells, results could be in the form of the molecules and structures that constitute the individual cells.

Is there an overall framework that might be adopted to organise this bacteria-based language? The concept of *competitive coherence* might help here (Norris 1998). Biological systems on all scales are confronted with the challenge of obtaining a future state that is coherent with environmental conditions and with previous states. These states are created by the active functioning of a set of constituents of the system. This active set is selected from the larger set available to the system. Many social organisations are constrained by the need to reconcile coherence with their present environment and coherence with their past environments. To grow and survive, research laboratories, for example, have to select an active set of workers in response to new discoveries and to new funding initiatives, but must reconcile this selection with the research history of the laboratory and, in particular, with its skills, experience and interests. To grow and survive, bacteria must also select an *active set* of macromolecules in response to external and internal conditions. Such responses entail both the generation of a coherent cell state, in which the cell's contents work together efficiently and harmoniously, and the generation of a coherent sequence of cell states. Contradiction and incoherence are punished since, for example, a cell that simultaneously induces the expression of genes for growth at high temperature and at low temperature is likely to be out-competed by rival cells that induce each set of genes only when needed. A cell that proceeded from one cell state to another very different one (without good environmental reason) would be wasting precious resources. A strong selective pressure therefore exists to generate active sets of constituents to provide both coherent cell states and a coherent sequence of such states. We have proposed that competitive coherence is responsible for generating these active sets (Norris 1998). This concept describes how a system maintains both the continuity of the composition of its active set via a Next process and the coherence of this active set (with respect to the inside and outside world) via a Now process (see "Beyond

high level instructions" section). Competitive coherence is a scale-free concept that operates at levels ranging from macromolecular assemblies to social groups (the nature of the Next and Now processes varies with the level). It has some features in common with concepts such as synergies (Haken 1983; Kelso 2008), SOWAWN machines (Ji 2009) and neural Darwinism (Edelman 1987).

How might competitive coherence constitute the unifying concept needed for the framework of a new language? Is it possible, for example, that the phenomenon of emergence corresponds to an active subset of constituents having an unexpected coherence with strong selective advantages (Norris et al. 2005)? Suppose each type of macromolecule has a large number of characteristics (as in the case of mRNA and proteins which contain a large number of sites that can bind water, ions, molecules and other macromolecules). As proteins are being chosen via competitive coherence to work together, suppose that the first ones to be chosen just happen to contain a binding site to the same molecule. Suppose that, in some environments, this combination of proteins proves useful. Suppose too that this molecule becomes available, perhaps for the first time. The presence of this binding site could then become an important factor in the coherence process which dominates the choice of the rest of the proteins to work together in the active set. In other words, the environment acts via the coherence process to lend importance to one out of many sites. The result is the selection of this site (plus the molecule that binds to it) as a determinant of the cell's response to a particular environment. More specifically, consider, for example, that (1) this binding site is for a particular phospholipid with long, saturated acyl chains, and (2) the proteins with this site bind to the phospholipid to form a domain in which they are juxtaposed and in which their activities complement one another. There might then be a selection for this binding site in other complementary proteins. In the language of competitive coherence, binding to this phospholipid would become a type of connectivity to determine membership of an active set and this active set would take on the physical form of a proteolipid domain responsible for a particular function. It is, therefore, conceivable that competitive coherence might act as a unifying concept for the myriad different connections and instructions that would lie at the heart of a new bacterial programming language.

Specific experiments

Travelling salesman

If bacteria are to be engineered to solve **NP** problems such as the travelling salesman as proposed above ("Solving the travelling salesman problem?" section), bacteria must be constructed that can move up a gradient of peptides towards the bacteria that are secreting them. Could such bacteria be made? The *agr* quorum-sensing system in Gram-positive staphylococci involves the secretion of a peptide, AgrD, in a modified form as an AIP (autoinducing peptide) of around seven to nine amino acids with a thiolactone ring (Novick and Geisinger 2008); this AIP can bind to a membrane-bound receptor, AgrC, which then phosphorylates the response regulator, AgrA, to activate transcription of the *agr* operon containing the *agrA*, *agrB*, *agrC* and *agrD* genes. Note that the ability of an AIP to activate its cognate receptor is highly sequence-specific and substitution of a single amino acid can change specificity. In principle the *agr* system might be modified to work in chemotaxis. First, an *agr* operon should be constructed that can operate in a model bacterium such as *B. subtilis* or even the Gram-negative *E. coli*. Second, this operon should be constructed so as to have an expression that is either constitutive or inducible. Third, a chemotaxis receptor such as McpB in *B. subtilis* should be modified by replacing its extramembrane sensing domain (which binds asparagine) with the domain from AgrC which binds the AIP. This step may require inactivation of other chemotaxis receptors which may nevertheless contribute to chemotaxis (Zimmer et al. 2002). It may also require mutagenesis of the modified *mcpB* and indeed other sequences bearing in mind that it would be possible to select those bacteria that successfully swim up a gradient of the AIP.

To implement the approach outlined in "Solving the travelling salesman problem?" section, a bacterium must be constructed that has input A and output B. More specifically, for example, a bacterium might be constructed that responds to one AIP (AIP_A) but that secretes another (AIP_B). This would entail taking say a *B. subtilis* with a modified chemotaxis receptor, McpB, that recognises AIP_A and introducing an operon such that it secretes a different AIP, AIP_B. The prediction is now that if the bacteria producing AIP_B were immobilized inside a capillary, a second set of bacteria with the cognate McpB for AIP_B would swim towards them (while themselves secreting AIP_A).

Bacteria must be then constructed in which the binding of the AIP is needed for the transcription of the gene encoding a macromolecule essential for growth. Initially, this might again be attempted by exploiting the *agr* system. In *Staphylococci*, activated AgrA also stimulates transcription of an RNA, RNAIII (which is itself a regulator) from the P3 promoter. This promoter has been fused to the gene encoding β-lactamase to measure AgrA activity (Novick and Geisinger 2008). Hence, the P3 promoter might be fused to a gene encoding a product essential for the growth of *B. subtilis* in conditions in which this gene

had been deleted from the chromosome (or had been mutated to give the bacterium a temperature sensitive phenotype). Suppose that the P3 promoter was fused to the gene encoding a tRNA. In this case, the bacterium could only grow in the presence of the AgrC cognate for the AIP secreted by another bacterium.

A series of bacteria must then be constructed of type A → B in which the rate of production of AIP_B is inversely proportion to the distance between the cities A and B. This could be achieved by different modifications to the *agr* operon used to produce AgrD, the precursor of AIP. For example, mutations in the promoter, which is bound by RNA polymerase, can reduce transcription, mutations in the Shine–Dalgarno sequence, which is a ribosomal binding site in the mRNA, upstream of the start codon AUG, can reduce translation, while alterations to mRNA and peptides can reduce their stability. This is not of course enough to solve even a toy example, since each bacterium must have a number of different AgrC receptors to bind the different AIPs equal to the number of cities minus one (see "Solving the travelling salesman problem?" section). Formation of active, heterologous receptors might then be a problem (Zimmer et al. 2002) requiring consideration. The experiments outlined above reveal a little of the nature of the challenge. Before embarking on the constructions needed, it would prudent to validate the concept using programmes for modelling cells that are based on realistic spatio-temporal parameters (for example, Amar et al. 2008), such validation for a small number of cities in silico might then pave the way for the larger number of cities that might, in principle, be attainable in vivo.

Encouragingly, bacteria have recently been used in a proof-of-principle experiment to solve an **NP**-complete problem, a Hamiltonian Path problem, in which a path must be found in a directed graph from the start node (such as a city in the salesman problem) to the end node that goes through each node just once. In the experiment (Baumgardner et al. 2009), this entailed considering each edge in a network as *half* of a gene connecting two nodes (with the exception of the start and end nodes); DNA recombination allowed these halves to be shuffled. This was achieved by having a *hixC* site at the end of each half gene so that in the presence of the Hin recombinase these sites could be recombined to reconstruct an ordered series of functional genes that corresponded to the solution. In this experiment, there was no selectable way to converge progressively on the solution although, as discussed above, this should not be too difficult.

The density problem

To begin to develop a bactoputer with the advantages of high density and adaptability, bacteria could be constructed in which only those bacteria grow that participate in the calculation. This could be achieved by encouraging non-participant bacteria to sporulate. Sporulation is regulated by an intricate network that relies primarily on the activity of three major transcriptional regulators: Spo0A (for sporulation), DegU (for degradation), and ComK (for competence) (Lopez et al. 2009). The idea would be to fuse a gene that is part of a circuit to a regulator, so that the regulator is produced (or is not produced) *only* if the circuit is active. In *B. subtilis*, for example, the gene that is part of the circuit could be fused to *abrB* which encodes the repressor of *sigH* which itself encodes an activator of Sp0A; care would have to be taken to prevent induction of competence and the production of matrix and to allow motility.

Optimisation and constraint problems in organisations

Access to information about phenotypic diversity is essential if it is to be used to solve problems. One way to achieve this would be via information on the proteome at the level of the individual cell using microarrays and recognition by antibodies. A technique that could prove precious here is SIMS given its capacity to detect, quantify and image (on the 100 nm scale) macromolecules such as antibodies (Dauphas et al. 2008).

Recognition

To develop a chip based on the transfer of molecules between different bacterial species, such transfer must be detected and preferably identified and quantified. One way to achieve this might be via a differential labelling of different bacterial species with stable isotopes followed by SIMS (Guerquin-Kern et al. 2005; Lechene et al. 2006; Legent et al. 2008). Bacteria can be labelled in vivo by growth on media enriched in relatively rare but stable isotopes such as ^{13}C and ^{15}N, etc.

Beyond high level instructions

To develop a framework for a set of instructions based on bacterial processes and constituents, we suggested above ("Beyond high level instructions" section) that the concept of competitive coherence might be helpful. Could evidence be obtained to support this? Competitive coherence can, in fact, be implemented in silico using conventional programming instructions. In such an implementation, the state of a system at time $n + 1$ is determined by a competition between the Next process, which is based on its state at time n, and the Now process, which is based on the developing $n + 1$ state itself (Norris 1998). Initial, small-scale experiments show that the programme can learn to

respond to a few environmental stimuli and even to distinguish between different patterns of these stimuli (Norris, unpublished). The question now is whether it can be scaled up into a programme that learns to respond to a large number of stimuli and that would serve as a model for the behaviour of the biological cell.

Discussion

It might be asked whether some aspects of bactoputing could be simulated by the latest generation of massively parallel computers. This would depend on the ambition. For a typical graphics processing unit, the stochastic automaton, HSIM, takes 1 min of computer time to simulate 1 min of bacterial time for a simplified version of glycolysis in a cell, irrespective of its size, containing a total of 1,000 molecules (or enzymes), irrespective of the number of classes of these molecules (Amar et al. 2004). A real bacterium contains several million proteins (Ishihama et al. 2008) (as well as lipids, RNA, metabolites and other molecules, plus inorganic ions), many of which are required for growth and survival; moreover, these molecules are required in spatially and temporally distinct structures (Llopis et al. 2010; Lopez and Kolter 2010). If each bacterium in a population were simulated by a single graphics processing unit (which is designed for parallel processing), the largest of such machines would be insufficiently powerful insofar as there would be about three orders of magnitude between the time taken for a GPU to simulate a million molecules in HSIM plus five orders of magnitude between the 10^8 to 10^9 bacteria in an millilitre and the largest computers containing 1,600 graphics processing units. Hence, a bactoputing approach that depends on growth and survival of individual bacteria within a population of bacteria cannot at present be simulated in silico (other than at a very simple level). In other words, a complete simulation of a colony of bacteria is not computationally feasible with present technology, and perhaps not even in principle as biological evolution may well have found the simplest physical implementation of the complex abilities of a living cell. However, it may well be possible that certain essential, but abstract features of bactoputing can be identified that can be simulated in a standard computer. This would be analogous to the paradigm of swarm intelligence where abstract principles underlying the solution of complex problems by ant colonies have been identified (Bonabeau et al. 1999; Jost et al. 2010). These principles can then be readily implemented in a computer to tackle such problems as the travelling salesman one, without having to simulate the full biological complexity of real ants. Since bacteria are much simpler (insofar as they are much smaller) than ants, this may well be possible for bactoputing as well. Thus, this line of research might ultimately lead to the identification of new general principles of parallel computation. That said, to the extent that bacteria are 'simpler' than ants, we may also stand a much higher chance of fully understanding the biological processes underlying the dynamics of bacterial colonies than the ones of ant populations, even though the former are by many orders of magnitude larger than the latter in terms of population size. Perhaps, then, a formalisation of bactoputing then will be more similar to a neural network as a scheme that abstracts the collective computational abilities of neurons in brains. In contrast to neural networks, in such bacterial networks, the communication between different elements will not work through specific pair couplings (synapses), but rather through more unspecific spatial diffusion of signals. (Note though that real bacteria can also have specific couplings when they conjugate.) This may offer the advantage of a much larger variety of different signals and logical functions and a much greater ability for flexible reorganisation. Specificity would thus not be achieved by a relatively fixed architecture as in neural networks, but rather through the specificity of signal receptors in the cells. However, such more abstract principles are not the main issue of the present paper, and we now return to the discussion of how actual biological colonies of bacteria can solve computational problems.

Computer scientists are interested in solving combinatorial problems of the **NP-complete** and related classes. We have suggested above that such problems make a good test case for bactoputing and that the autocatalytic growth properties of bacterial populations might be exploited to solve the travelling salesman problem. This is an illustration of a weak form of bactoputing that, arguably, is just bioputing, since it is not really in the nature of bacteria to perform the task required here. Another weak form of bactoputing would entail constructing bacteria with their metabolic enzymes on the outside (an 'inside-out' metabolism) to create a heterogeneous population in which each individual bacterium needs the activity of other bacteria to grow. A form of bactoputing that is stronger than the inside-out example—but related to it—would be to make use of those bacteria like *Clostridium cellulovorans* that use cellulsomes to degrade the walls of plants and that naturally have metabolic activities on the outside (Doi and Kosugi 2004).

Computer scientists are also interested in solving problems with hardware, and here bacterial populations offer huge densities (a human intestinal tract contains up to 10^{14} bacteria) with numerous chemical and physical connections in three dimensions (Matsuhashi et al. 1998; Palkova 2004). They are also cheap and grow fast as well as being robust and self-repairing. Some species can operate at high

temperatures and, of course, the presence of water is not a problem.

In the world of human affairs, an organisation often has to steer between survival and growth where conflicting constraints make it hard to find good solutions. A possible approach to finding these solutions is to use one complex system to model another. Bacteria have been selected for billions of years for their capacity to explore phenotype space; this entails exploiting opportunities to grow and to survive stresses. Both opportunities and challenges come in a huge number of combinations in an evolutionary landscape that is modified by the behaviour of the bacteria themselves. Here, we have suggested that it is in bacterial solutions to the challenge of navigating phenotype space that bactoputer scientists may discover new paradigms and applications. For example, bacterial populations anticipate nutritional crashes and, in doing this, they communicate with one another and lyse (Kolodkin-Gal et al. 2007), they also increase phenotypic diversity in the rundown to stationary phase (Vohradsky and Ramsden 2001). The prediction here is that as oil supplies run out and global warming increases our societies will go through a period of experimentation, which if unsuccessful, will be followed by convergence on some spartan model. Maybe a bactoputer could help us do this intelligently.

The use of bacterial colonies in their native state as 'brains' to solve recognition and other problems would be a strong form of bactoputing, as it would be based on such natural properties of bacteria as chemotaxis and morphological change. Understanding, learning and memory are hallmarks of neural function and would be captured by the spatiotemporal distribution of bacterial species and phenotypes, possessing distinct fluorescent labels, within a stack of transparent, etched chips in response to changing inputs (as discussed above in the case of *P. dendritiformis* and *P. vortex*). A quantifiable learning would then correspond to the migration and growth of bacteria with particular phenotypes in appropriately structured regions of the stack, memory would correspond to the persistence of these bacterial distributions and understanding would correspond to the relationship between these distributions and input patterns. This stack might therefore be described as a bacterial brain or bactoputer (see "Recognition and other problem" section). Encouragingly, *E. coli* reveals a pattern-forming ability within artificial mazes (with channels in the tens of microns) based on production of and chemotaxis towards glycine (Park et al. 2003; Park et al. 2003).

Another strong form of bactoputing could directly involve the metabolism (the network of reactions, catalysed by enzymes, that creates the cell) and, given that metabolic enzymes are encoded by genes, it should be possible to design circuits based on coupling metabolism and gene expression (Thellier et al. 2006). Such bactoputing could be useful in studying social systems where money is both a 'nutrient' and a signal. Significantly, it has been shown that a regulatory circuit in metabolism, namely the lycopene biosynthesis pathway in *E. coli*, can be engineered to control gene expression in response to the intracellular metabolite, acetyl phosphate (Farmer and Liao 2000).

Perhaps the most exciting aspect of bactoputing would lie in the development of a totally new high language for computing based on the language that bacteria themselves speak (divide, replicate DNA, mutate, lyse, produce phage, conjugate, etc.). As our understanding of regulation in bacteria increases, our capacity to manipulate bacteria—to give them instructions—also increases. But if we can speak to them, can we also listen? For that, we need better access to the phenotypes of individual bacteria and better understanding of the processes that generate these phenotypes. Technological advances in 'omics' will one day—perhaps soon via such techniques as SIMS (see "Optimisation and constraint problems in organisations" section)—give rapid access to information on the proteomes, phosphorylomes, lipidomes, interactomes, metabolomes, etc., of large numbers of individual cells. A better understanding of biological processes could also come from new concepts of biological origin. Bactoputer scientists should be looking for them now.

Acknowledgements We thank the Epigenomics Project, Genopole® and the Network of Excellence, Nanobeams, for support. We thank Kepa Ruiz Mirazo, Martyn Amos, Paul Bourgine and anonymous referees for incisive comments and Philippe Marliere for several ideas.

Appendix

Here, we review for computer scientists some well-known facts about bacteria and mention some recent speculations with the idea that these may be useful for bactoputing. We then propose population-based approaches in which the bacterial population is a single computer but in which each bacterium is a different computing device.

Phenotypic diversity

Population diversity can pose a problem to those types of bactoputing that require bacteria to behave in standard, constant ways. One solution to this lies in constructing negative feedback circuits to limit the range over which the

concentrations of network components fluctuate, as shown for simple genetic circuits in *E. coli* (Becskei and Serrano 2000). However, population diversity can be seen as a solution in search of a problem. One of these problems is species extinction where a possible solution would lie in preserving the small sub-groups in which a disproportionate fraction of the diversity is concentrated (Rauch and Bar-Yam 2004).

Rather than thinking bacteria as identical individuals, it is often useful to think of them in terms of populations of heterogeneous individuals that compete, collaborate and communicate. Bacteria already use peptides and other chemicals that they export into the media, and that they then sense to determine population density in the phenomenon of quorum sensing which is implicated in processes that include symbiosis, virulence, competence, conjugation, antibiotic production, motility, sporulation, and biofilm formation (Miller and Bassler 2001).

Many bacteria are confronted with the problem of a changing environment in which different and sometimes incompatible strategies are required for survival and for growth. This is resolved at the population level by the generation of both phenotypic diversity (Tolker-Nielsen et al. 1998; Booth 2002; Balaban et al. 2004; Avery 2006) and genetic diversity. In generating phenotypic diversity, transcription factors are clearly important and, since they are often present in small numbers, there is a role to be played by stochastic noise (Elowitz et al. 2002; Sato et al. 2003); however, the key role, we and others have argued, is played by the cell cycle (Norris and Madsen 1995) which leads to the presence of two or more chemically identical chromosomes within the same cytoplasm that spontaneously adopt complementary patterns of expression to equip the future daughter cells for life in different environments (Minsky et al. 2002). In generating genetic diversity, there is an interesting phenomenon whereby certain individuals in a stressed population undergo mutations in proof-reading genes that lead to a high level of mutations; when these unhealthy individuals lyse, fragments of their DNA can be taken up and used by other individuals which may thus acquire a beneficial mutation (Matic et al. 2004). It turns out that for environmental stresses to induce programmed cell death in cultures of *E. coli*, the bacteria must secrete a specific pentapeptide that is derived from the degradation of glucose-6-phosphate dehydrogenase, a metabolic enzyme (Kolodkin-Gal et al. 2007). This may mean that there is yet another connection to be understood between individual metabolism and population signalling.

Plasmids, bacteriophage and transposons

Bacteria possess small 'chromosomes' or plasmids that are replicated independently of their principal chromosome and that can be transferred readily between bacteria. Genetic information can also be transferred between bacteria by bacteriophages; these bacterial viruses are stable and resistant and, protected by a shell of proteins, often transport DNA from one bacterium to another. Genetic information can be transferred within a chromosome or between a chromosome and a plasmid by transposons. This allows them to adapt to exposure to new dangers and to avail themselves of new opportunities. Hence bacteria possess a powerful armoury for altering and rearranging their genetic material. They possess, in other words, a system for both solving problems and for anticipating problems.

Hyperstructures

In the pursuit of the nature of the bacterial cell, we and others have explored the possibility of the existence of a level of organisation intermediate between macromolecules and whole cells—the level of hyperstructures (Amar et al. 2002; Guzman et al. 2002; Molina and Skarstad 2004). A hyperstructure is a collection of diverse molecules (genes, mRNAs, proteins, ions, lipids) that is associated with at least one function. A non-equilibrium hyperstructure is assembled into a large, spatially distinct structure to perform a function and is disassembled, wholly or partially, when no longer required (Norris et al. 2004). Examples in *E. coli* of non-equilibrium hyperstructures include a nucleolar hyperstructure (analogous to the microcompartment within which ribosomes are assembled inside the eukaryotic nucleus) for synthesising ribosomal RNA (Cabrera and Jin 2003), the dynamic cytoskeletal hyperstructures and the division hyperstructure responsible for the invagination of the membrane and peptidoglycan layer (Aarsman et al. 2005). An equilibrium hyperstructure is also a large spatially distinct structure with a function, but its life is not dependent on spending energy. Examples, again in *E. coli*, include the highly ordered RecA-DNA co-crystal, which forms when there is insufficient ATP to repair DNA damage, and in which the tight crystalline packaging is believed to protect the DNA by physically sequestering it (Levin-Zaidman et al. 2000). Certain hyperstructures straddle the non-equilibrium/equilibrium divide such as the flagellar hyperstructure which has both an equilibrium part (the flagellum itself) and, during the formation of the flagellum, a non-equilibrium part comprising the transcribed genes and their products that acts as a sensor of hydration (Wang et al. 2005).

In the hyperstructure approach, hyperstructures are responsible for structuring membranes, cytoplasm and nucleoid, and a state cycle of hyperstructures is responsible for progress through the cell cycle. Such hyperstructures may interact via a variety of mechanisms including the

familiar processes of DNA supercoiling, coupled transcription/translation, molecular and macromolecular signalling, tensegrity and local concentrations, as well as the speculative ones of ion condensation, oscillating water structures and intracellular streaming.

Minimal genomes

Bacteria have existed for billions of years. As the growing problem of antibiotic resistance shows, they readily adapt to and escape from human control. Computers based on bacteria are therefore likely to have a short lifespan unless the adaptability of bacteria is taken into account or indeed unless it becomes part of the computing. One approach to make bacteria more malleable is to take bacteria such as *E. coli* and *Bacillus subtilis* and to cut down the genome so as to eliminate 'unnecessary' functions (Kobayashi et al. 2003; Gil et al. 2004). How far might this be taken? Until recently, it was believed that 250 or so genes would be needed for a minimal version of a modern cell in the most favourable conditions (Mushegian and Koonin 1996) similar to minimal genome sizes inferred by site-directed gene disruptions and transposon-mediated mutagenesis knockouts in several bacteria (for references, see Luisi et al. 2006). However, the symbiont *Carsonella ruddii*, which lives inside insects, has a 160 kb chromosome that encodes only 182 proteins although admittedly it does lack many genes that are thought essential for independent life outside a host (Nakabachi et al. 2006). Hence, the number of 250 genes might be reduced considerably, for example, if repair and other functions are dispensed with and if protein synthesis is imagined to be performed with a reduced set of ribosomal proteins. Attempts to generate bacteria with minimal genomes have led to engineered *E. coli* strains with nearly 30% of the genome missing, certain of which grow more slowly than the wild-type strain (Hashimoto et al. 2005). More recently, elimination of recombinogenic sequences and mobile DNA (such as transposons and IS elements), as well as elimination of 'non-essential' and cryptic functions, have generated strains of *E. coli* that have increased genomic stability, maintain otherwise unstable plasmids and can be electroporated readily with DNA (Posfai et al. 2006); moreover, these strains grow well. While such strains may be less prone to discarding or perverting the constructs that scientists have inserted into them, they may be less able to follow the natural strategies of bacteria based on the generation of genetic and phenotypic diversity (see above), strategies that may be either welcome or unwelcome depending on the type of computing to be performed.

In developing bacteria for computing purposes, little use has yet been made of L-forms. These are bacteria that have been selected for the loss of their peptidoglycan wall. Despite this major change, these simplified bacteria manage to grow and divide (Onoda et al. 2000; Siddiqui et al. 2006). Although fragile, they seem to have cytoplasmic membranes that are naked, and they may be easier to manipulate and may be more amenable for computing than their parent bacteria. In a sense, they represent a step back towards earlier forms of life that could usefully undergo genome shrinking.

Directed evolution

Another approach of possible value to bactoputing is that of directed evolution. Mutators have defective DNA proofreading and generate mutations at a high frequency. Such mutators can be grown for thousands of generations in chemostats under a constant selective pressure to drive genotype and phenotype towards those desired by the experimenter. These selective conditions can result in the bacteria adapting in ways that are not desired, for example, by increasing their capacity to stick the walls of the chemostat and so avoid being flushed out; this type of problem, which may arise in both the construction of the strain and in the operation of the bacteria-based computer, can be partly resolved with two chemostats one being used while the other is being sterilised (de Crecy-Lagard et al. 2001). The mutations in such conditions occur independently of one another in individual bacteria. However, if a proportion of bacteria lyse, the possibility exists that other bacteria can take up their DNA and benefit (see above). Hence, the rapidity of directed evolution can be increased by the use of bacterial species that take up foreign DNA at a high frequency such as *Acinetobacter* sp. ADP1 (Palmen et al. 1993).

Liposomes and origins of life

There are two ways in which study of the origins of life may be useful for a bacteria-based computing. One is in the investigation of what the first cell really was (assuming it really was a specific cell rather than a population or ecosystem Hunding et al. 2006); clearer ideas about the nature of the first cells might help us in exploiting their descendants. The second is in the experiments performed. A prime example of combining such hypotheses with experiments is in the construction of the minimal cell de novo. This is a bottom-up approach (as opposed to the top-down approach of deleting chunks from an existing genome described above) where the objective is to generate the simplest cell that can be considered alive (Luisi et al. 2006). The definition of 'alive' here need not trouble those whose objective is to obtain devices for computing. One concept is that of a minimal RNA cell comprising a vesicle with two ribozymes inside, the first of which catalyses the

synthesis of the components that self-assemble into membrane, while the second replicates both itself and the first ribozyme (Szostak et al. 2001). The properties of vesicles (liposomes when their constituents are lipids) continue to be intensively explored. Vesicles can grow using surfactant precursors and even divide to maintain the original size distribution (for references, see Luisi et al. 2006). Heterogeneous composition and the presence of channels may help circumvent the problem of the difficult entry of materials in modern membranes based on phospholipids or similar molecules. For example, an α-heamolysin pore incorporated into liposomes permits the uptake of small metabolites from the medium (Noireaux et al. 2005; Noireaux and Libchaber 2004). A very different way of bringing ions and indeed macromolecules into liposomes might be based on the channels formed by the simple compounds polyphosphate and poly-β-hydroxybutyrate (Das et al. 1997; Norris 2005). An alternative, population-based, approach would be to develop the fusion and fission of heterogeneous liposomes (Norris and Raine 1998). There have been numerous experiments on fusion of compartments using water-in-oil emulsions which have the advantage of allowing high local concentrations of reactants (for references, see Luisi et al. 2006). Often, these reconstruction approaches entail the production of proteins detectable by fluorescence, such as the green fluorescent protein (GFP). A mutant form of GFP has, for example, been produced in lecithin liposomes (Yu et al. 2001). Continued development in this area towards real minimal cells coupled to detection systems may prove useful for computing. The work in the PACE project is relevant here (http://complex.upf.es/∼ricard/PACEsite).

The vision in the bottom-up construction of minimal cells is that they should contain a small set of macromolecules with highly specific functions: the original cells started out simple and became complex (Luisi et al. 2006). This vision is being fleshed out experimentally (Luisi et al. 2006). A very different vision is that life appeared in the form of a pre-biotic ecology in which a rich, diverse and complex world of protocells or composomes exchanged their contents (Hunding et al. 2006). In this vision, the first cells only have meaning within the context of a population and to investigate and exploit this, the minimal cell must give way to the minimal population.

Colonies and swarming

Populations of the bacterium, *Paenibacillus dendritiformis*, make surfactants to extract fluid from the semi-solid nutrient substrate, so as to create a layer within which they can swim. The problem is that the production of the surfactant requires the collective action of a dense bacterial population which the food-depleted substrate cannot sustain. The solution they have adopted is to form a colony with a branching structure, within each branch the bacterial density is sufficiently high, yet the average population density of the colony is sufficiently low for the nutrients to suffice. Very different patterns form at different nutrient levels. Part of the solution resides in the precise adjustment of the viscosity of the lubricant layer and the production rate of the surfactant in order to generate specific branch structures with specific widths according to the substrate hardness and nutrient levels (Kozlovsky et al. 1999; Ben-Jacob and Levine 2005). *P. dendritiformis* growing on poor substrates can have either a branching (B) or chiral (C) morphology. On hard substrates where high densities are required to produce enough lubricating fluid, the B morphotype is selected, leading to the formation of colonies with branching, bush-like morphologies whilst on softer substrates, the C morphotype is selected, leading to curly branches that allow faster expansion while also using patches of food left behind as the branches are twisted inward. How exactly are the branches made? Cells go into a non-motile state further back from the colony front, where the nutrient levels are extremely low. They also emit quorum-sensing molecules or pheromones that represent the state of the population and its environment and that occasion changes in gene expression. One of these changes is the inhibition of cell division which leads to them elongating. Upon elongation, the cells alter their collective movement from the typical run-and-tumble of the short B cells to a coordinated forward–backward movement that leads to the branches twisting with a specified handedness (this handedness depends on cell–cell interactions together with the inherent flagella handedness). The two possible morphotypes are inheritable and can coexist for some range of growth conditions. There are also spontaneous transitions to give new patterns that maximise the rate of colony expansion.

Learning from experience has also been described in bacteria. *Paenibacillus vortex* forms vortices that vary in size from tens to millions of bacteria, according to their location in the colony. The cells in the vortex replicate, and the vortex expands in size and moves outward as a unit, leaving behind a trail of motile but usually non-replicating cells—the vortex branch. Maintaining the integrity of the vortex while it serves as a higher order building block of the colony requires communication: each cell in the vortex needs to be informed that its role is now more complex, being a member of both the specific vortex and the whole colony, so it can adjust its activities accordingly. This ongoing communication is particularly apparent when it comes to the birth of new vortices. New vortices emerge in the trail behind a vortex following initiation signals that cause the bacteria there to produce more lubricating fluid and to move quite rapidly as a turbulent "biofluid", until an

eddy forms and becomes a new vortex. The entire process appears to proceed as a continuous dialogue: a vortex grows and moves, producing a trail of bacteria and being pushed forward by the very same bacteria left behind. At some point the process stalls, and this is the signal for the generation of a new vortex behind the original one, that leaves home (the trail) as a new entity towards the colonisation of new territory. Recent findings based on *P. vortex* and other bacteria indicate that bacteria modify their colonies in the presence of antibiotics so as to optimise bacterial survival. It also appears that these bacteria have a short-term memory which enables them to recall the structural solution they found to the antibiotic to which they were exposed most recently (Ben-Jacob et al. 2004).

Signalling

Within bacterial and other cells, there are numerous types of signalling pathways of relevance to computing (Bray 1990). These include the well-studied two component pathways (Baker and Stock 2007; Laub and Goulian 2007) and other systems (Grangeasse et al. 2007) that depend on phosphorylation, those that depend on alarmones such as ppGpp (Wang et al. 2007), systems that depend on poly-β-hydroxybutyrate (Das et al. 1997; Norris 2005) and those that depend on ions (perhaps even on ion condensation Ripoll et al. 2004). Another conceptually very different class of signals exists, at least potentially, in cells. This is the class generated by those enzymes that only associate with one another when they are actively engaged in catalysing their cognate reactions (so giving rise to functioning-dependent structures); a wide variety of types of signals in the form of enzymes or metabolites can be generated (Thellier et al. 2006).

Within bacterial communities, chemical signalling occurs via molecules such as N-acyl-homoserine lactones for Gram-negative bacteria, post-translationally modified peptides for Gram-positive bacteria and furanosyl-borate diester for all species (Palkova 2004) as well as fragments of intracellular enzymes (Kolodkin-Gal et al. 2007). There has also been intriguing evidence for physical signalling in bacterial communication (Matsuhashi et al. 1996; Norris and Hyland 1997).

References

Aarsman ME, Piette A, Fraipont C, Vinkenvleugel TM, Nguyen-Disteche M et al (2005) Maturation of the *Escherichia coli* divisome occurs in two steps. Mol Microbiol 55:1631–1645

Adamatzky A, de Lacy Costello B (2002) Collision-free path planning in the Belousov-Zhabotinsky medium assisted by a cellular automaton. Naturwissenschaften 89:474–478

Adleman LM (1994) Molecular computation of solutions to combinatorial problems. Science 266:1021–1024

Amar P, Ballet P, Barlovatz-Meimon G, Benecke A, Bernot G et al (2002) Hyperstructures, genome analysis and I-cell. Acta Biotheor 50:357–373

Amar P, Bernot G, Norris V (2004) HSIM: a simulation programme to study large assemblies of proteins. J Biol Phys Chem 4:79–84. http://www.lri.fr/~pa/Hsim/JBPC_04.pdf

Amar P, Legent G, Thellier M, Ripoll C, Bernot G et al (2008) A stochastic automaton shows how enzyme assemblies may contribute to metabolic efficiency. BMC Syst Biol 2:27

Atkinson MR, Savageau MA, Myers JT, Ninfa AJ (2003) Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in *Escherichia coli*. Cell 113:597–607

Avery SV (2006) Microbial cell individuality and the underlying sources of heterogeneity. Nat Rev Microbiol 4:577–587

Baker M (2005) Better living through microbes. Nat Biotechnol 23:645–647

Baker MD, Stock JB (2007) Signal transduction: networks and integrated circuits in bacterial cognition. Curr Biol 17:R1021–R1024

Balaban NQ, Merrin J, Chait R, Kowalik L, Leibler S (2004) Bacterial persistence as a phenotypic switch. Science 305:1622–1625

Barabasi AL, Oltvai ZN (2004) Network biology: understanding the cell's functional organization. Nat Rev Genet 5:101–113

Basu S, Gerchman Y, Collins CH, Arnold FH, Weiss R (2005) A synthetic multicellular system for programmed pattern formation. Nature 434:1130–1134

Baumgardner J, Acker K, Adefuye O, Crowley ST, Deloache W et al (2009) Solving a Hamiltonian Path Problem with a bacterial computer. J Biol Eng 3:11

Becskei A, Serrano L (2000) Engineering stability in gene networks by autoregulation. Nature 405:590–593

Ben-Jacob E, Levine H (2005) Self-engineering capabilities of bacteria. J R Soc Interface. doi:10.1098/rsif.2005.0089

Ben-Jacob E, Shapira Y (2004) Meaning-based natural intelligence vs. information-based artificial intelligence. In: Ben-Nun H (ed) The cradle of creativity. Shaarei Tzedek, Jerusalem

Ben-Jacob E, Becker I, Shapira Y, Levine H (2004) Bacterial linguistic communication and social intelligence. Trends Microbiol 12:366–372

Benenson Y, Gil B, Ben-Dor U, Adar R, Shapiro E (2004) An autonomous molecular computer for logical control of gene expression. Nature 429:423–429

Ben-Jacob E (2003) Bacterial self-organization: co-enhancement of complexification and adaptability in a dynamic environment. Philos Trans A Math Phys Eng Sci 361:1283–1312

Bonabeau E, Dorigo M, Theraulaz G (1999) Swarm intelligence. Oxford University Press, Oxford, p 320

Booth IR (2002) Stress and the single cell: intrapopulation diversity is a mechanism to ensure survival upon exposure to stress. Int J Food Microbiol 78:19–30

Bray D (1990) Intracellular signalling as a parallel distributed process. J Theor Biol 143:215–231

Bulter T, Lee SG, Wong WW, Fung E, Connor MR et al (2004) Design of artificial cell-cell communication using gene and metabolic networks. Proc Natl Acad Sci USA 101:2299–2304

Cabrera JE, Jin DJ (2003) The distribution of RNA polymerase in *Escherichia coli* is dynamic and sensitive to environmental cues. Mol Microbiol 50:1493–1505

Carbone A, Seeman NC (2002) Circuits and programmable self-assembling DNA structures. Proc Natl Acad Sci USA 99:12577–12582

Conrad M (1995) Scaling of efficiency in programmable and non-programmable systems. Biosystems 35:161–166

Das S, Lengweiler UD, Seebach D, Reusch RN (1997) Proof for a nonproteinaceous calcium-selective channel in *Escherichia coli* by total synthesis from (R)-3-hydroxybutanoic acid and inorganic phosphate. Proc Natl Acad Sci USA 94:9075–9079

Dauphas S, Delhaye T, Lavastre O, Corlu A, Guguen-Guillouzo C et al (2008) Localization and quantitative analysis of antigen-antibody binding on 2D substrate using imaging NanoSIMS. Anal Chem 80:5958–5962

de Crecy-Lagard VA, Bellalou J, Mutzel R, Marliere P (2001) Long term adaptation of a microbial population to a permanent metabolic constraint: overcoming thymineless death by experimental evolution of *Escherichia coli*. BMC Biotechnol 1:10

Demarse TB, Wagenaar DA, Blau AW, Potter SM (2001) The neurally controlled Animat: biological brains acting with simulated bodies. Auton Robots 11:305–310

Denning PJ, Comer DE, Gries D, Mulder MC, Tucker A et al (1989) Computing as a discipline. Commun ACM 32:9–23

Doi RH, Kosugi A (2004) Cellulosomes: plant-cell-wall-degrading enzyme complexes. Nat Rev Microbiol 4:541–551

Edelman G (1987) Neural Darwinism. The theory of neuronal group selection. Basic Books, New York

Elowitz MB, Levine AJ, Siggia ED, Swain PS (2002) Stochastic gene expression in a single cell. Science 297:1183–1186

Engel GS, Calhoun TR, Read EL, Ahn TK, Mancal T et al (2007) Evidence for wavelike energy transfer through quantum coherence in photosynthetic systems. Nature 446:782–786

Farmer WR, Liao JC (2000) Improving lycopene production in *Escherichia coli* by engineering metabolic control. Nat Biotechnol 18:533–537

Fröhlich H (1978) Coherent electric vibrations in biological systems and the cancer problem. IEEE Trans Microw Theory Tech MTT-S 26:613–617

Gardner TS, Cantor CR, Collins JJ (2000) Construction of a genetic toggle switch in *Escherichia coli*. Nature 403:339–342

Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-completeness. Freeman W.H., New York

Gil R, Silva FJ, Pereto J, Moya A (2004) Determination of the core of a minimal bacterial gene set. Microbiol Mol Biol Rev 68: 518–537

Grangeasse C, Cozzone AJ, Deutscher J, Mijakovic I (2007) Tyrosine phosphorylation: an emerging regulatory device of bacterial physiology. Trends Biochem Sci 32:86–94

Grover LK (1997) Quantum mechanics helps in searching for a needle in a haystack. Phys Rev Lett 79:325–328

Guerquin-Kern JL, Wu TD, Quintana C, Croisy A (2005) Progress in analytical imaging of the cell by dynamic secondary ion mass spectrometry (SIMS microscopy). Biochim Biophys Acta 1724:228–238

Guzman EC, Caballero JL, Jimenez-Sanchez A (2002) Ribonucleoside diphosphate reductase is a component of the replication hyperstructure in *Escherichia coli*. Mol Microbiol 43:487–495

Haken H (1983) Synergetics, an introduction: Nonequilibrium phase transitions and self-organization in physics, chemistry and biology. Springer, Berlin

Hashimoto M, Ichimura T, Mizoguchi H, Tanaka K, Fujimitsu K et al (2005) Cell size and nucleoid organization of engineered *Escherichia coli* cells with a reduced genome. Mol Microbiol 55:137–149

Hunding A, Kepes F, Lancet D, Minsky A, Norris V et al (2006) Compositional complementarity and prebiotic ecology in the origin of life. BioEssays 28:399–412

Ingber DE (1998) The architecture of life. Sci Am 278:30–39

Ishihama Y, Schmidt T, Rappsilber J, Mann M, Hartl FU et al (2008) Protein abundance profiling of the *Escherichia coli* cytosol. BMC Genomics 9:102

Janniere L, Canceill D, Suski C, Kanga S, Dalmais B et al (2007) Genetic evidence for a link between glycolysis and DNA replication. PLoS ONE 2:e447

Ji S (2009) Molecular theories of the living cell: conceptual foundations molecular mechanisms and applications. Springer, New York

Jost J, Bertschinger N, Olbrich E (2010) Emergence. A dynamical systems approach. New Ideas Psychol 28:265–273

Kauffman S (1996) At home in the Universe the search for the laws of complexity. Penguin, London, pp 1–321

Kelso JAS (2008) Synergies: atoms of brain and behavior. In: Sternad D (ed) A multidisciplinary approach to motor control. Springer, Heidelberg

Kobayashi K, Ehrlich SD, Albertini A, Amati G, Andersen KK et al (2003) Essential *Bacillus subtilis* genes. Proc Natl Acad Sci USA 100:4678–4683

Kolodkin-Gal I, Hazan R, Gaathon A, Carmeli S, Engelberg-Kulka H (2007) A linear pentapeptide is a quorum-sensing factor required for mazEF-mediated cell death in *Escherichia coli*. Science 318:652–655

Kozlovsky Y, Cohen I, Golding I, Ben-Jacob E (1999) Lubricating bacteria model for branching growth of bacterial colonies. Phys Rev E 59:7025–7035

Laub MT, Goulian M (2007) Specificity in two-component signal transduction pathways. Annu Rev Genet 41:121–145

Laub MT, McAdams HH, Feldblyum T, Fraser CM, Shapiro L (2000) Global analysis of the genetic network controlling a bacterial cell cycle. Science 290:2144–2148

Lechene C, Hillion F, McMahon G, Benson D, Kleinfeld AM et al (2006) High-resolution quantitative imaging of mammalian and bacterial cells using stable isotope mass spectrometry. J Biol 5:20

Legent G, Delaune A, Norris V, Delcorte A, Gibouin D et al (2008) Method for macromolecular colocalization using atomic recombination in dynamics SIMS. J Phys Chem B 112:5534–5546

Levin-Zaidman S, Frenkiel-Krispin D, Shimoni E, Sabanay I, Wolf SG et al (2000) Ordered intracellular RecA-DNA assemblies: a potential site of in vivo RecA-mediated activities. Proc Natl Acad Sci USA 97:6791–6796

Llopis PM, Jackson AF, Sliusarenko O, Surovtsev I, Heinritz J et al (2010) Spatial organization of the flow of genetic information in bacteria. Nature 466:77–81

Lopez D, Kolter R (2010) Functional microdomains in bacterial membranes. Genes Dev 24:1893–1902

Lopez D, Vlamakis H, Kolter R (2009) Generation of multiple cell types in *Bacillus subtilis*. FEMS Microbiol Rev 33:152–163

Luisi PL, Ferri F, Stano P (2006) Approaches to semi-synthetic minimal cells: a review. Naturwissenschaften 93:1–13

Matic I, Taddei F, Radman M (2004) Survival versus maintenance of genetic stability: a conflict of priorities during stress. Res Microbiol 155:337–341

Matsuhashi M, Pankrushina AN, Endoh K, Watanabe H, Ohshima H et al (1996) *Bacillus carboniphilus* cells respond to growth-promoting physical signals from cells of homologous and heterologous bacteria. J Gen Appl Microbiol 42:315–323

Matsuhashi M, Pankrushina AN, Takeuchi S, Ohshima H, Miyoi H et al (1998) Production of sound waves by bacterial cells and the response of bacterial cells to sound. J Gen Appl Microbiol 44:49–55

Miller MB, Bassler BL (2001) Quorum sensing in bacteria. Annu Rev Microbiol 55:165–199

Minsky A, Shimoni E, Frenkiel-Krispin D (2002) Stress, order and survival. Nat Rev Mol Cell Biol 3:50–60

Molina F, Skarstad K (2004) Replication fork and SeqA focus distributions in *Escherichia coli* suggest a replication

hyperstructure dependent on nucleotide metabolism. Mol Microbiol 52:1597–1612
Mushegian AR, Koonin EV (1996) A minimal gene set for cellular life derived by comparison of complete bacterial genomes. Proc Natl Acad Sci USA 93:10268–10273
Nakabachi A, Yamashita A, Toh H, Ishikawa H, Dunbar HE et al (2006) The 160-kilobase genome of the bacterial endosymbiont carsonella. Science 314:267
Noireaux V, Libchaber A (2004) A vesicle bioreactor as a step toward an artificial cell assembly. Proc Natl Acad Sci USA 101:17669–17674
Noireaux V, Bar-Ziv R, Godefroy J, Salman H, Libchaber A (2005) Toward an artificial cell based on gene expression in vesicles. Phys Biol 2:P1–P8
Norris V (1998a) Bacteria as tools for studies of consciousness. In: Hameroff S, Kaszniak A, Scott A (eds) Toward a science of consciousness II: the second Tucson discussions and debates. MIT Press, Cambridge, pp 397–405
Norris V (1998b) Modelling *E.* coli: the concept of competitive coherence. Comptes Rendus de l'Academie des Sciences 321:777–787
Norris V (2005) Poly-(R)-3-hydroxybutyrate and the pioneering work of Rosetta Natoli Reusch. Cell Mol Biol (Noisy-le-grand) 51:629–634
Norris V, Hyland GJ (1997) Do bacteria "sing"? Mol Microbiol 24:879–880
Norris V, Madsen MS (1995) Autocatalytic gene expression occurs via transertion and membrane domain formation and underlies differentiation in bacteria: a model. J Mol Biol 253:739–748
Norris V, Raine DJ (1998) A fission-fusion origin for life. Origins Life Evol Biosph 28:523–537
Norris V, Amar P, Bernot G, Delaune A, Derue C et al (2004) Questions for cell cyclists. J Biol Phys Chem 4:124–130
Norris V, Cabin A, Zemirline A (2005) Hypercomplexity. Acta Biotheor 53:313–330
Norris V, Janniere L, Amar P (2007) Hypothesis: variations in the rate of DNA replication determine the phenotype of daughter cells. In: Amar P, Képès F, Norris V, Bernot G (eds) EDP sciences. Genopole, Evry, pp 71–81
Novick RP, Geisinger E (2008) Quorum sensing in *Staphylococci*. Annu Rev Genet 42:541–564
Onoda T, Enokizono J, Kaya H, Oshima A, Freestone P et al (2000) Effects of calcium and calcium chelators on growth and morphology of *Escherichia coli* L-form NC-7. J Bacteriol 182:1419–1422
Ozbudak EM, Thattai M, Lim HN, Shraiman BI, van Oudenaarden A (2004) Multistability in the lactose utilization network of *Escherichia coli*. Nature 427:737–740
Palkova Z (2004) Multicellular microorganisms: laboratory versus nature. EMBO Rep 5:470–476
Palmen R, Vosman B, Buijsman P, Breek CK, Hellingwerf KJ (1993) Physiological characterization of natural transformation in *Acinetobacter calcoaceticus*. J Gen Microbiol 139:295–305
Park S, Wolanin PM, Yuzbashyan EA, Lin H, Darnton NC et al (2003a) Influence of topology on bacterial social interaction. Proc Natl Acad Sci USA 100:13910–13915
Park S, Wolanin PM, Yuzbashyan EA, Silberzan P, Stock JB et al (2003b) Motion to form a quorum. Science 301:188
Pfaffmann JO, Conrad M (2000) Adaptive information processing in microtubule networks. Biosystems 55:47–57
Posfai G, Plunkett G III, Feher T, Frisch D, Keil GM et al (2006) Emergent properties of reduced-genome *Escherichia coli*. Science 312:1044–1046
Possoz C, Filipe SR, Grainge I, Sherratt DJ (2006) Tracking of controlled *Escherichia coli* replication fork stalling and restart at repressor-bound DNA in vivo. EMBO J 25:2596–2604
Rauch EM, Bar-Yam Y (2004) Theory predicts the uneven distribution of genetic diversity within species. Nature 431:449–452
Ripoll C, Norris V, Thellier M (2004) Ion condensation and signal transduction. BioEssays 26:549–557
Rocha E, Fralick J, Vediyappan G, Danchin A, Norris V (2003) A strand-specific model for chromosome segregation in bacteria. Mol Microbiol 49:895–903
Rothemund PW, Papadakis N, Winfree E (2004) Algorithmic self-assembly of DNA Sierpinski triangles. PLoS Biol 2:e424
Sato K, Ito Y, Yomo T, Kaneko K (2003) On the relation between fluctuation and response in biological systems. Proc Natl Acad Sci USA 100:14086–14090
Siddiqui RA, Hoischen C, Holst O, Heinze I, Schlott B et al (2006) The analysis of cell division and cell wall synthesis genes reveals mutationally inactivated ftsQ and mraY in a protoplast-type L-form of *Escherichia coli*. FEMS Microbiol Lett 258:305–311
Skretas G, Wood DW (2005) A bacterial biosensor of endocrine modulators. J Mol Biol 349:464–474
Szostak JW, Bartel DP, Luisi PL (2001) Synthesizing life. Nature 409:387–390
Thellier M, Legent G, Amar P, Norris V, Ripoll C (2006) Steady-state kinetic behaviour of functioning-dependent structures. FEBS J 273:4287–4299
Thomas R (1980) On the relation between the logical structure of systems and their ability to generate multiple steady states or sustained oscillations. Ser Synerg 9:180–193
Tolker-Nielsen T, Holmstrom K, Boe L, Molin S (1998) Non-genetic population heterogeneity studied by in situ polymerase chain reaction. Mol Microbiol 27:1099–1105
Tsuda S, Zauner K-P, Gunji Y-P (2006) Robot control: from silicon circuitry to cells. In: Ijspeert AJ, Masuzawa T, Kusumoto S (eds) BioADIT 2006. Lecture Notes in Computer Science. Springer, Heidelberg, pp 20–32
Van Regenmortel MHV (2004) Emergence in biology. In: Amar P, Comet J-P, Kepes F, Norris V (eds) Modelling and simulation of biological processes in the context of genomics. Genopole, Evry, pp 123–132
Vohradsky J, Ramsden JJ (2001) Genome resource utilization during prokaryotic development. FASEB J 15:2054–2056
Wall ME, Hlavacek WS, Savageau MA (2004) Design of gene circuits: lessons from bacteria. Nat Rev Genet 5:34–42
Wang Q, Suzuki A, Mariconda S, Porwollik S, Harshey RM (2005) Sensing wetness: a new role for the bacterial flagellum. EMBO J 24:2034–2042
Wang JD, Sanders GM, Grossman AD (2007) Nutritional control of elongation of DNA replication by (p)ppGpp. Cell 128:865–875
You L, RSr Cox, Weiss R, Arnold FH (2004) Programmed population control by cell-cell communication and regulated killing. Nature 428:868–871
Yu W, Sato K, Wakabayashi M, Nakaishi T, Ko-Mitamura EP et al (2001) Synthesis of functional protein in liposome. J Biosci Bioeng 92:590–593
Zemirline A, Norris V (2007) Non-separability, multiple interactions and hypercomplexity. In: Amar P, Kepes F, Norris V, Bernot G (eds) EDP sciences. Evry, France, pp 99–102
Zimmer MA, Szurmant H, Saulmon MM, Collins MA, Bant JS et al (2002) The role of heterologous receptors in McpB-mediated signalling in *Bacillus subtilis* chemotaxis. Mol Microbiol 45:555–568