Nat Comput (2011) 10:119–127
DOI 10.1007/s11047-010-9184-2

Biocircuit design through engineering bacterial logic gates

Angel Goñi-Moreno · Miguel Redondo-Nieto · Fernando Arroyo · Juan Castellanos

Published online: 20 March 2010

Abstract Designing synthetic biocircuits to perform desired purposes is a scientific field that has exponentially grown over the past decade. The advances in genome sequencing, bacteria gene regulatory networks, as well as the further knowledge of intraspecies bacterial communication through quorum sensing signals are the starting point for this work. Although biocircuits are mostly developed in a single cell, here we propose a model in which every bacterium is considered to be a single logic gate and chemical cell-to-cell connections are engineered to control circuit function. Having one genetically modified bacterial strain per logic process would allow us to develop circuits with different behaviors by mixing the populations instead of re-programming the whole genetic network within a single strain. Two principal advantages of this procedure are highlighted. First, the fully connected circuits obtained where every cellgate is able to communicate with all the rest. Second, the resistance to the noise produced by inappropriate gene expression. This last goal is achieved by modeling thresholds for input signals. Thus, if the concentration of input does not exceed the threshold, it is ignored by the logic function of the gate.

A. Goñi-Moreno (✉)
Grupo de Computación Natural, Facultad de Informática, Universidad Politécnica de Madrid, 28660 Madrid, Spain
e-mail: agmoreno@gcn.upm.es

M. Redondo-Nieto
Depto. Biología, Facultad de Ciencias, Universidad Autónoma de Madrid, 28049 Madrid, Spain
e-mail: miguel.redondo@uam.es

F. Arroyo
Depto. de Lenguajes, Proyectos y Sistemas Informáticos, Escuela Universitaria de Informática, Universidad Politécnica de Madrid, 28031 Madrid, Spain
e-mail: farroyo@eui.upm.es

J. Castellanos
Artificial Intelligence Department, Facultad de Informática, Universidad Politécnica de Madrid, Campus de Montegancedo, Boadilla del Monte s/n, 28660 Madrid, Spain
e-mail: jcastellanos@fi.upm.es

A. Goñi-Moreno · F. Arroyo · J. Castellanos
Natural Computing Group, Universidad Politécnica de Madrid, Madrid, Spain

Keywords Synthetic biology · Logic gate · Biocircuit · Cell–cell communication

1 Introduction

This new century has seen the emergence of a new research area between computer science and biology called synthetic biology. The aims of this new field are the design of not existing biological devices or systems and the re-design of existing natural systems for programmed and desired purposes (Alon 2003). By manipulating the regulatory activities that cells perform and controlling gene expression, scientist have focused their attention on the development of devices similar to logic gates (Weiss et al. 1998; Storz et al. 2009; Nyan et al. 2008) with different features. The engineering of new biological components based on an analogy with computers will influence many other scientific disciplines and the resulting devices could be used in different applications involving smart drug modeling or ecology developments.

Although simple logic gates have been successfully programmed inside living cells, the objective of making complex circuits is still a main goal in synthetic biology. All the attempts to build difficult gene networks have found problems mainly due to the fuzzy behavior of the processes in the cell and the disturbances such as gene expression background noises (Chen et al. 2009). The stochastic nature of the processes in living cells which can cause those intrinsic noises are many, such as promoter background activity, unspecific transcription factor binding, protein degradation, transcriptional, and translational rates (Hooshangi and Weiss 2006). Taking all into account, it is an arduous task to control the input and the output of every logic gate in a complex circuit implemented in vivo.

As cooperative skills between cell-gates are engineered to control circuit function, here we propose a model to design logic circuits in which noise attenuation is achieved by programming thresholds for input responses.

The advances in genome sequencing as well as in transcriptome analysis are allowing the building of databases focused on gene regulatory networks (Seshasayee et al. 2006; Janky et al. 2009). These networks are relaying on the presence of cell sensors that can trigger a signal cascade, which modulates the activity of gene transcription factors and hence activation or repression of gene transcription.

It has been recently shown the existence of a communication system among bacteria named Quorum Sensing (QS) using small molecules among others the *N*-acylhomoserine lactone (AHL) family (Fuqua et al. 1994; Eberhard et al. 1981).

This system is being studied worldwide as allows bacteria coordination that leads to the formation of complex biological structures and social communities such as biofilms (Costerton et al. 1995, 1999). It has been found a big amount of bacteria strains using this communication system, mainly proteobacteria. The latter use orthologs to the Lux system detected in bioluminiscent symbiotic bacteria *Vibrio fischeri* (Engebrecht et al. 1983, 1984; Romero-Campero et al. 2008). AHLs are synthesized by LuxI and small differences in its sequence and therefore in its activity provoke slight variations in the AHL molecule, which is recognized by its specific LuxR. Thus, there are a huge amount of pairs LuxI-luxR that make quorum sensing a very active research topic in synthetic biology, as signal transmitters among bacteria.

This paper is organized as follow: Sect. 2 introduces the idea of using the fundamental chemical processes in living cells in order to produce logic functions such AND, OR and

NOT. It will also be presented in Sect. 2 our proposal to make the cell a self-contained single gate, as well as the benefits that creating those cells with the essential genes will have in order to design reusable circuit components. In Sect. 3, we explain in detail an example of a circuit build following the model of one gate per bacterial strain and input thresholds. Some features of the system are highlighted like scalability, reusability and modularity. The paper ends with the conclusions of Sect. 4.

2 Structural elements of biocircuits: cellular logic gates

In order to build a nanodevice or engineered biocircuit that performs logic operations by using bacteria, it is important to define from the beginning the structural elements that are useful for our developments. Those elements are the logic gates needed to design more complex circuits. Logic cells must be able to process one or more inputs and express an output according to the function they have been genetically programmed to carry out.

The design of logic-cells is based on changing the natural transcriptional mechanisms to suit logic functions. The cell performs important regulatory activities trough DNA-binding proteins called transcription factors and genomic regions preceding genes called gene promoters, which are the targets for transcription factors. The structure and sequence of gene promoters define their nature. Briefly, we can find constitutive promoters that don't need any external influence to activate gene transcription, and promoters whose activity is regulated by transcription factors that can act as activators or repressors. The first ones regulates one or more genes by increasing the rate of transcription whereas the latter ones fulfill the regulation by decreasing that rate. Furthermore, cells can regulate transcription factors activities depending on external signals that can activate or repress transcription factor activities. Among those signals we can highlight AHLs, nutrients, and abiotic factors as temperature.

Databases as RegulonDB (http://regulondb.ccg.unam.mx/) (Huerta et al. 1998), DBD (http://dbd.mrc-lmb.cam.ac.uk/DBD/index.cgi?Home) (Wilson et al. 2008) or PRODORIC (http://prodoric.tu-bs.de/) (Münch et al. 2003), are storing data appearing in the different genomic and transcriptomic projects. Thus, nowadays a lot of transcription factors and promoter sequences are available to build functional units to be fused to another that will be used to generate new receptors, input or output signals. Therefore, on manipulating all those biological units, we can engineer living cells in order to force them to act with a logic behavior.

Circuit design on living cells, like other fields in synthetic biology, is a research area that depends heavily upon the advances done by molecular biology. New data are acquired daily that help us to understand the regulatory processes of cells. The knowledge obtained from the insights into the cell mechanisms has increased exponentially over the last decade. It is difficult to believe that the first complete genome sequenced was obtained less that 15 years ago. Nowadays, research focus on bacterial genome sequencing and annotation of functional units and it is revealing the presence of neutral sites where insertion of new functional units or logical gates without affecting cell functionality and viability. Moreover, lately a lot of studies are trying to find out the minimal genome content that a bacteria can afford without losing their capabilities to feed themselves, grow and duplicate as reviewed by Moya (2009). In this work we propose the idea of encoding each logic function in a single bacteria, either inserting the "gene functions" in the neutral sites of the chromosome or designing de novo a minimal microorganism with the new information. Both options would emphasize the reusable possibilities that those cells would have. In

fact, if we had each gate programmed in a different bacterial strain we could combine them to form separated circuits instead of programming the whole circuit in a single cell anytime a different function would be needed.

A great diversity of operations can be performed with bacterial gates. That is due to the fact that genetic operations can work in an analogical way. They are not forced, like conventional computers are, to work only digitally. Lately, it is becoming a strong research area in Computer Science to try to develop analog logic gates and analog circuits. However, only a simulation of an analog behavior via fuzzy logic and probabilistic decisions is possible to carry out in silico. We could implement an algorithm using a few analog gates rather that the thousands required in digital operations for the same task. The great benefit of biocircuits based on living cells concerning this feature is that the analog behavior is found intrinsically in them. That is why the attention should not be focused only in the development of strict logic cell-gates but also in the design of gates with real analog functioning.

In Fig. 1 some of the strict and analog logic gates that are possible to design using cells as processing units can be found so that they can be used as the structural elements when building a biocircuit.

Figure 1(1) represents a basic implication. If the bacteria receives molecules *a*, it returns molecules *d*. Figure 1(2) and (3) are AND logic bacteria. Only if the bacteria receives both molecules can return the output. It is remarkable that the input molecules can be external to the cell (outside the membrane) or internal. The aim of using internal inputs is to design more complex circuits in which extra-cellular communication cooperates with intra-cellular regulatory circuits. Figure 1(4) represents an NAND gate, Fig. 1(5) an OR gate and Fig. 1(6) and inversor. Figure 1(7) shows the possibility of designing strict logic bacteria with multiple input.

In the analog logic section of Fig. 1, there are two graphics that shows the relation between the input signals and the reporter intensity (light intensity in case of using *Luciferase*) in the output. The first image represents a inversely proportional function: the higher the input is, the lower output we get. In contrast, the other image shows a directly proportional function: the higher the input is, the higher output we obtain. This cell-gates encode real analogical behavior.

3 Biocircuit community

The similarities between computers and bacteria communities are many since we can engineer the last ones to perform desired functions, in other words, to execute complete computations. Logic circuits are hardware devices that consists of a number of logic gates which are able to process data and perform logical operations on it. In the case of designing logic biocircuits over bacteria communities, the so called biohardware is formed by the bacteria themselves and the surface that holds them.

From the time this technology appeared (Knight and Sussman 1997) a big effort is done to build increasingly difficult circuits in bacteria (Hinze et al. 2008; Danchin 2009; Brenner et al. 2007; Batt et al. 2007). In the model described here, gates are implemented using one genetically modified bacteria strain for each in order to highlight the computational power of intercellular communication, parallelism, reusability and noise attenuation. The logic-cells exhibit cooperativity by modelling quorum signals between them like it is shown next.

The main advantages of this procedure are found in the characteristics of the connections between the inputs/outputs of one cell-gate and the inputs/outputs of another. There are two remarkable features in that linkage:

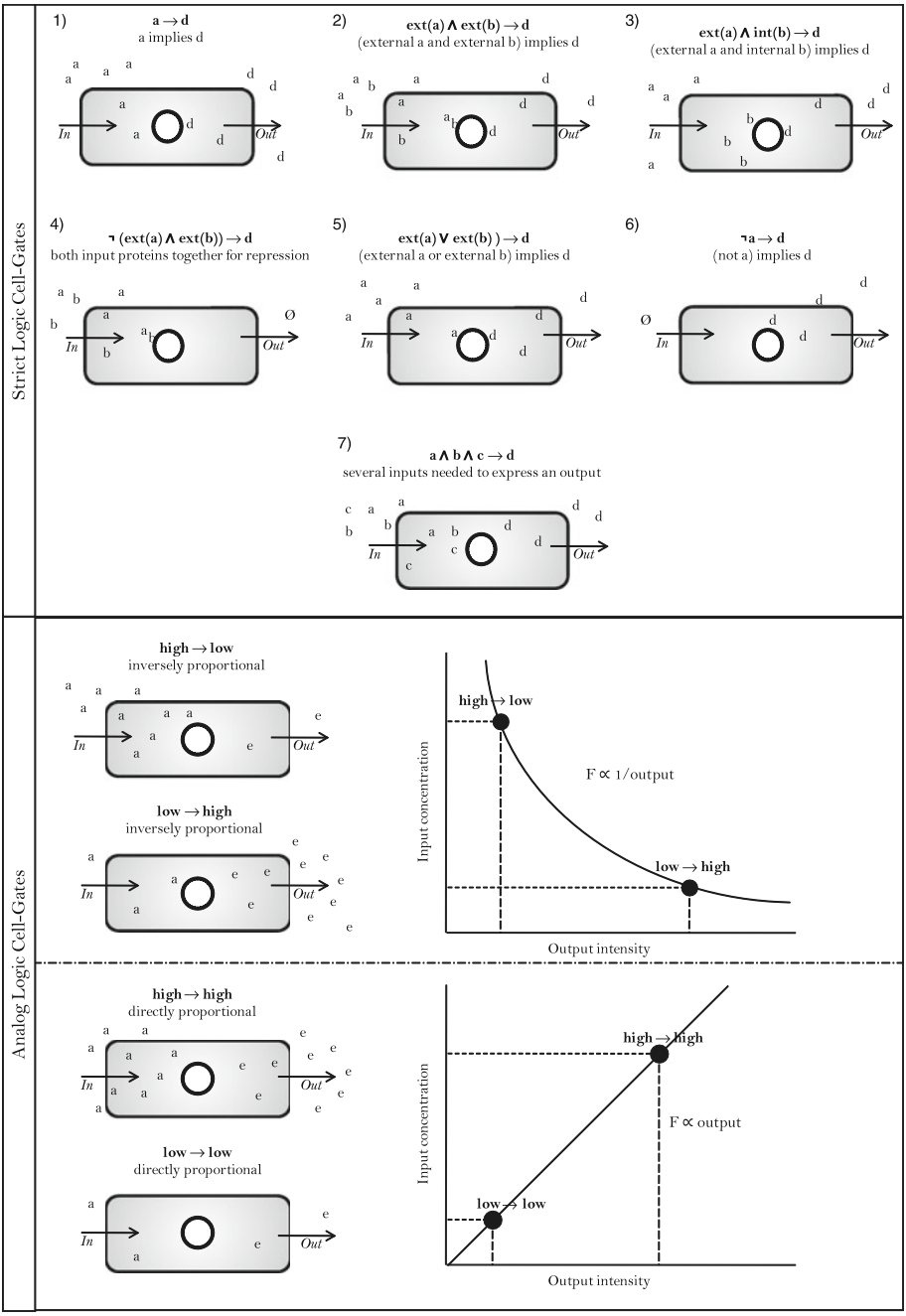

Fig. 1 Logic and analog cell-gates. (*a*, *b*, *c*) represent signal inputs (i.e. AHLs) that induce gene expression. (*d*) output molecules can represent either a signal molecule like (*a*, *b*, *c*) or a reporter protein (Luciferase, GFP). (*e*) are only output reporters for final responses

- *Concurrent signals*. Fully connected circuits appear by mixing togheter more than one cell strain with different configurations (one logic function per strain). That is due to the fact that in a community with N gates, $Com = [B_0 . . .B_N]$, the output of one gate B_i can be evaluated by the rest of the community in parallel: $B_{i\_out} \rightarrow B_{n\_in}$ where $n = 0, 1, \ldots, i-1, i+1, \ldots, N$.
- *Noise-attenuant thresholds*. In an attempt to reduce noise in gene expression, this type of network uses regulatory thresholds for input signals. Every gate B_i in the community has a concrete threshold δ assigned to its input. The δ parameter indicates the minimum level of signals needed in order to produce a response in the target cell. If the output produced by B_i must be used as the input of B_j, and the number of signal molecules is denoted by $|x|$ we could affirm that when $|B_{i\_out}| \geq \delta_{B_j}$ the logic fuction of B_j is activated.

Communication between gates takes place in an intermediate agent that plays the role of a buffer, or a shared memory, in a reader/writer problem. This buffer is the nutrient solution in which the cells are in, through which signals are transmitted along the circuit. Every bacteria *reads* the nutrient solution permanently using its cell membrane and as soon as it finds the suitable inputs to activate its logic fuction, it *writes* in the buffer the output molecules. It is in the *read* process where a threshold is defined in order to reduce the noise due to inappropriate input concentrations. Approximately there are 2×10^9 cell-gates per millilitre all reading/writing simultaneously.

All that computational features are used here to describe and design a biocircuit based on a three-strain community with XOR behavior. The XOR function, or Exclusive-OR, between two signals A and B is defined as either A or B, but not both. Therefore, output response must answer this statement: NOT A and B, or A and NOT B. Then the final function for this biocircuit is $\overline{A}B + A\overline{B} = A \oplus B$.

Figure 2 shows the translation of the logic abstraction [Fig. 2(1)] into the device in vivo implemented with bacterial gates [Fig. 2(2)]. This circuit contains three different kinds of gates: OR [Fig. 2(2a)], NAND [Fig. 2(2b)] and AND [Fig. 2(2c)]. The final community will contain then three bacterial strains, one per gate. This circuit produces outputs which are strictly dependent from the inputs given thus a *combinatorial* logic circuit is designed by this method. There are two architectural levels in the circuit: the high-level where the processors (each bacterial gate) are considered as black-boxes and it does not use knowledge of the internal structure of them, and the low-level architecture which specifies the functioning of the gates like in a white-box test.

Analyzing the circuit from the high-level point of view there could be observed five input/output signals. Arabinose and IPTG are the initial inputs of the problem. Those inputs must fire the cascade that finally results in a XOR circuit. Both of them are read by the bacterial gates of Fig. 2(2a) and (2b), which depending on the absence (input = 0) and presence (input = 1) of each molecule returns a different response. These responses or outputs are molecules of the family AHL_i where $i = \{a, b\}$. When the NAND and OR bacteria produce their output "writing" it in the nutrient solution, the AND bacteria receives those signals and fire their internal function that leads the circuit to achieve the final solution.

The low-level design of the bacterial gates can be graphically seen in Fig. 2—2 and it works as follow:

- OR gates express *luxIa* anytime an inducible promoter is active. That happens whenever Arabinose which activates pAraBAD, or IPTG which activates pLac are present as inputs.

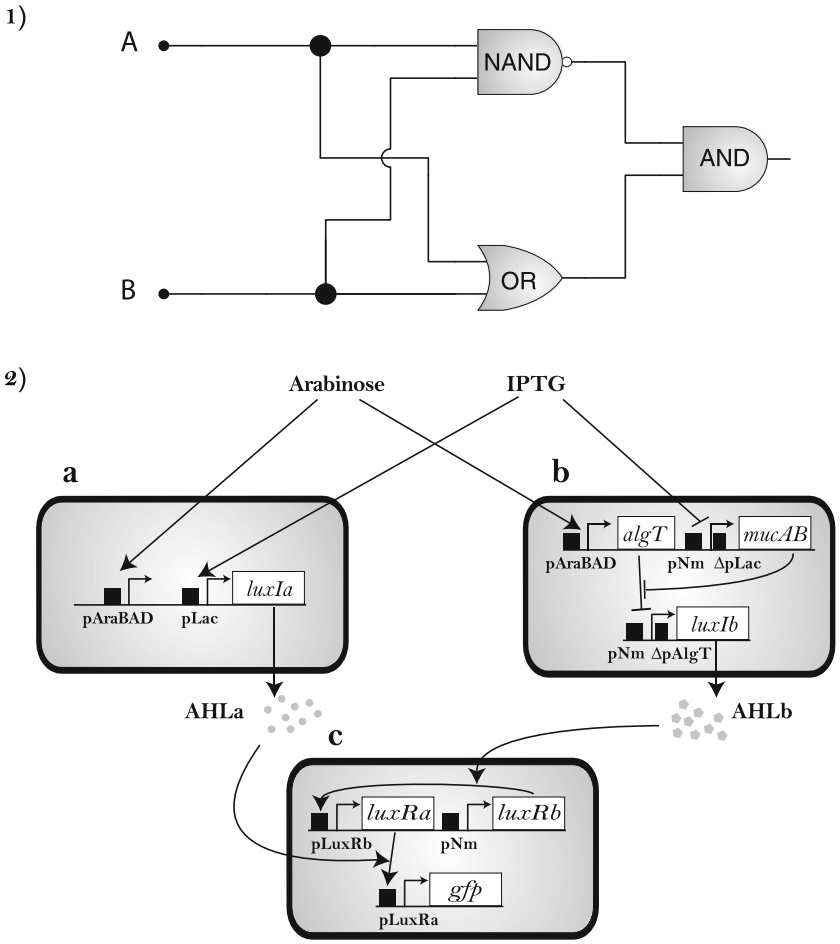

Fig. 2 Proposal for a XOR circuit design. *1* schematic representation of a XOR circuit, *2* biological design of a XOR circuit using 3 transgenic bacteria—*2a* OR bacterial gate, *2b* NAND bacterial gate, *2c* AND bacterial gate. *pAraBAD* arabinose inducible promoter, *pLac* IPTG inducible promoter, Δ*pLac* truncated IPTG inducible promoter, Δ*pAlgT* truncated *algT* promoter, *pNm* neomycin constitutive promoter, *gfp* green fluorescent protein

– NAND gates express *luxIb* anytime that both inputs are not present at the same time. Neomycin promoters (pNm) which induce constantly the expression of downstream genes, i.e. *mucab* and *luxib*, unless they are blocked by the presence of transcription factors binding to the downstream truncated promoters, are the clue of the design of these gates. The constant expression of system MucAB posttranscriptionally blocks any AlgT transcription factor activity (Mathee et al. 1997). Thus, these gates stop expressing *luxIb*, and hence producing AHLb, if and only if both Arabinose and IPTG are *true* inputs. IPTG deactivates the promoter pNmΔpLac, blocking the production of any *mucab* that might trap Arabinose induced AlgT which represses the complex pNmΔpAlgT.

– AND gates receive AHL_i as inputs from the previous gates and control the expression of the final output response (*gfp*). Constant expression of *luxRb* prepares a pool of receptors waiting for the appearance of AHLb in order to induce the expression of AHLa sensors, LuxRa. Thus, if and only if both AHL_i are present, the LuxRa is active and triggers the expression of *gfp*.

The resulting distributed net has two different layers: an input layer which is formed by NAND and OR gates and an output layer formed by AND gates. It is important to notice that all the bacteria in the output layer start functioning when a concrete concentration of AHL_i is achieved. That threshold is in charge of avoiding noise by ignoring lower concentration of input molecules.

4 Conclusions

Bacterial computing is an emerging area that is being improved by the appearance of more and more developments every year. In the future it will become easier to design biodevices due to the Synthetic Biology community's attempt to join the efforts (http://partsregistry.org) and the exponential improvements in biological processes knowledge and molecular biology technologies.

The biocircuits implemented in vivo by engineering bacterial communities are normally designed using only one bacterial strain where billions of cells are programmed in the same way. However, in nature there are biofilms (Jayaraman and Wood 2008) in which hundreds of bacterial species constitute a robust and stable community by improving their cooperative skills. This work pretends to reach that capacity by engineering communication among stains for desire purposes. One of the main problems of the proposed system is to synchronize the growth of different bacterial strains in order to avoid a bias in the algorithm. Also, future works must try to find the suitable substrate where the bacteria communities shall be kept. Nevertheless, there are already some studies describing several organic (hydrogels) and inorganic (sol–gel) substrates to stabilize bacteria, reviewed by Bjerketorp (2006), and laser methods to print this bacteria matrices onto surfaces (Barron et al. 2004).

Generating bacterial logic gates can lead to solve new problems by mixing different engineered transgenic bacterial strains. For example, the XOR circuit designed in previous sections makes up a three-strain community. As interaction between strains is performed by quorum sensing signals, a common problem of biocircuits arising from stochastic expression events can be faced by engineering input sensitivity or thresholds in the receptors of bacterial gates.

Finally, storing the logic instructions in the chromosome of the cells makes unnecessary using episomes to keep the information. The fact that this method avoids the use of resistance systems like antibiotics, reduces metabolic pressures on the bacteria and diminishes horizontal gene transfer processes among different bacterial strains that could originate short circuits, are clear advantages of this biodevice.

Acknowledgement M. Redondo-Nieto is the recipient of a contract from MICROAMBIENTE-CM Consortium Program from Comunidad Autónoma de Madrid (Spain).

References

Alon U (2003) The Tinkerer as an engineer science. Biolog Netw 301(5641):1866–1867

Barron JA, Rosen R, Jones-Meehan J, Spargo BJ, Belkin S, Ringeisen BR (2004) Biological laser printing of genetically modified *Escherichia coli* for biosensor applications. Biosens Bioelectron 20(2):246–252
Batt G, Yordanov B, Weiss R, Belta C (2007) Robustness analysis and tuning of synthetic gene networks. Syst. Biol 23(18):2415–2422
Bjerketorp J, Hkansson S, Belkin S, Jansson JK (2006) Advances in preservation methods: keeping biosensor microorganisms alive and active. Curr Opin Biotechnol 17(1):43-49
Brenner K, Karig DK, Weiss R, Arnold FH (2007) Engineered bidirectional communication mediates a consensus in a microbial biofilm consortium. Proc Natl Acad Sci USA 104(44):17300–17304
Chen BS, Chang CH, Lee HC (2009) Robust synthetic biology design: stochastic game theory approach. Bioinformatics 25(14):1822–1830
Costerton JW, Lewandowski Z, Caldwell DE, Korber DR, Lappin-Scott HM (1995) Microbial biofilms. Annu Rev Microbiol 49:711–745
Costerton JW, Stewart PS, Greenberg EP (1999) Bacterial biofilms: a common cause of persistent infections. Science. 284:1318–1322
Danchin A (2009) Bacteria as computers making computers. FEMS Microbiol Rev 33(1):3–26
Eberhard A, Burlingame AL, Eberhard C, Kenyon GL, Nealson KH, Oppenheimer NJ (1981) Structural identification of autoinducer of *Photobacterium fischeri* luciferase. Biochemistry 20(9):2444–2449
Engebrecht J, Silverman M (1984) Identification of genes and gene products necessary for bacterial bioluminescence. Proc Natl Acad Sci USA 81:4154–4158
Engebrecht J, Nealson K, Silverman M (1983) Bacterial bioluminescence: isolation and genetic analysis of functions from *Vibrio fischeri*. Cell 32:773–781
Fuqua WC, Winans SC, Greenberg EP (1994) Quorum sensing in bacteria: the LuxR-LuxI family of cell density-responsive transcriptional regulators. J Bacteriol 176:269–275
Hinze T, Lenser T, Matsumaru N, Dittrich P, Hayat S (2008) Biosignal-based computing by AHL induced synthetic gene regulatory networks. From an in vivo flip-flop implementation to programmable computing agents. HFSP J 2(5):239–243
Hooshangi S, Weiss R (2006) The effect of negative feedback on noise propagation in transcriptional gene networks. Chaos 16(2):26108
Huerta AM, Salgado H, Thieffry D, Collado-Vides J (1998) RegulonDB: a database on transcriptional regulation in *Escherichia coli*. Nucleic Acids Res 26(1):55–59
Janky R, van Heiden J, Madan Babu M (2009) From analysis of complex networks to discovery of cis-regulatory elements Methods. Investig Transcr Regul 48(3):277–286
Jayaraman A, Wood TK (2008) Bacterial quorum sensing: signals, circuits, and implications for biofilms and disease. Annu Rev Biomed Eng 10:145–167
Knight TF, Sussman GJ (1997) Cellular gate technology. MIT Artificial Intelligence Laboratory, Cambridge
Mathee K, McPherson CJ, Ohman DE (1997) Posttranslational control of the *algT* (*algU*)-encoded sigma22 for expression of the alginate regulon in *Pseudomonas aeruginosa* and localization of its antagonist proteins MucA and MucB (AlgN). J Bacteriol 179(11):3711–3720
Moya A, Gil R, Latorre A, Juli Peret J, Garcillán-Barcia MP, de la Cruz F (2009) Toward minimal bacterial cells: evolution vs design. FEMS Microbiol. Rev 33:225–235
Münch R, Hiller K, Barg H, Heldt D, Linz S, Wingender E, Jahn D (2003) PRODORIC: prokaryotic database of gene regulation. Nucleic Acids Res 31(1):266–269
Nyan M, Smolke ChD (2008) Higher-Order cellular information processing with synthetic RNA devices. Science 322(5900):456–460
Romero-Campero FJ, Perez-Jimenez MJ (2008) A model of the quorum sensing system in *Vibrio fischeri*. Artif Life 14(1):95–109
Seshasayee ASN, Bertone P, Fraser GM, Luscombe NM (2006) Transcriptional regulatory networks in bacteria: from input signals to output responses Curr. Opin Microbiol 9(5):511–519
Storz G, Waters L (2009) Regulatory RNAs in bacteria. Cell 136(4):615–628
Weiss R, Homsy G, Nagpal R (1998) Programming biological cells. In: Proceedings of the 8th international conference on architectural support for programming languages and operating systems (ASPLOS-VIII), San Jose, California, 3–7 October 1998
Wilson D, Charoensawan V, Kummerfeld SK, Teichmann SA (2008) DBD–taxonomically broad transcription factor predictions: new content and functionality. Nucleic Acids Res 36(Database issue): D88–D92