FEBS Letters 582 (2008) 1237–1244

Minireview

Mining logic gates in prokaryotic transcriptional regulation networks

Rafael Silva-Rocha, Víctor de Lorenzo*

Centro Nacional de Biotecnología, CSIC, Campus de Cantoblanco, Madrid 28049, Spain

Received 10 January 2008; accepted 28 January 2008

Available online 12 February 2008

Edited by Patrick Aloy and Robert B. Russell

Abstract Prokaryotic transcriptional networks possess a large number of regulatory modules that formally implement many of the logic gates that are typical of digital, Boolean circuits. Yet, natural regulatory elements appear most often compressed and exaggeratedly context-dependent for any reliable circuit engineering barely comparable to electronic counterparts. To overcome this impasse, we argue that designing new functions with biological parts requires (i) the recognition of logic gates not yet assigned but surely present in the meta-genome, (ii) the *orthogonalization* and *disambiguation* of natural regulatory modules and (iii) the development of ways to tackle the connectivity and the definition of boundaries between minimal biological components.

Keywords: Digital circuit; Logic gate; Catabolic regulator; Metagenome; Orthogonalization

1. Introduction

One of the trademarks of Synthetic Biology is the rational combination of regulatory modules in artificial circuits for performing non-natural tasks, including complex binary computation operations based on logic gates [1,2]. The basis of such an endeavour is the implicit adoption of the metaphor of the cell as a sort of Turing machine. In this way, physicochemical environmental signals (the *inputs*) activate an existing gene expression *program* (encoded in the DNA), which is ultimately executed by transcriptional regulators on promoters and then by the downstream protein expression machinery [3]. This results, e.g. in changes of the cell metabolism through the increase or decrease of the production rate of specific proteins (the *output*). Under this conceptual frame, the program behind any biological function could in principle be de-constructed into minimal operative units, called by many *biological parts* (see http://parts.mit.edu [1]). Such units can then ideally be re-assembled following a rational blueprint to perform a different program, resulting in altogether new properties and behaviours. In this respect, *Synthetic Biology* clearly takes off from what since the late 1970s was called *Genetic Engineering*, as it brings into Biology robust engineering principles such as abstraction, hierarchical design, modularization and definition of systems boundaries – rather than vague analogies to cutting and pasting DNA sequences. In this mini-review, we briefly assess what is actually available for designing genetic circuits, how to upgrade natural modules to meet the requirements of robust engineering, and where to find the pieces that are still missing. Furthermore, we raise the questions of connectivity and evolvability of biological modules as two of the major bottlenecks that hinder the development of synthetic biological circuitry.

2. De-constructing naturally-occurring genetic circuits into usable regulatory elements

The principal actors of the biological input/output functions are the *cis*-(promoters) and the *trans*-regulatory elements (transcriptional regulators). Prokaryotic transcriptional factors (TFs) drive the activity of their cognate promoter(s) in response to one or more environmental stimuli. TFs can generally be *activators* by enhancing the binding or the activity of the RNA polymerase (RNAP) in the cognate promoters, or *repressors* by blocking this binding, or both [4]. Most known prokaryotic activators bind the upstream region of a promoter in response to a signal (for example, a substrate of the metabolic pathway regulated by the TF) and enhance the recruitment of the RNAP to the site. Alternatively, they may promote the escape and further progression of the transcription machinery from the promoter into the transcribed DNA sequence [5]. In contrast, transcriptional repressors typically interfere with the binding of RNAP to the −35 and −10 DNA hexamers of bacterial promoters. In this case, environmental stimuli decrease the affinity of the TF for its binding site, thereby allowing the RNAP to access the promoter and proceed with transcription [6,7]. One question relevant to circuit design emerges now: why activators and repressors instead of just one mechanism or the other? Sometimes the very same biological function (for instance, the *ara* systems for arabinose consumption) is positively regulated in one bacterium (*E. coli*, activated by AraC [8]) and negatively controlled in another (*B. subtillis*, repressed by AraR [9]). There is not an easy answer to this. It seems that activators generally produce more transcriptional output than repressors [10]. It is also likely that positive regulation allows a higher connectivity of the corresponding promoter to physiological co-regulation [11].

2.1. Prokaryotic promoters as Boolean logic gates

The participation of one or more TFs in the regulation of a given promoter confers the system the ability of integrating

*Corresponding author. Fax: +34 91 585 45 06.
E-mail address: vdlorenzo@cnb.csic.es (V. de Lorenzo).

0014-5793/$32.00
doi:10.1016/j.febslet.2008.01.060

different input signals in a fashion not unlike those described by the gates of Boolean logic. Such gates perform operations on one or more inputs and produce each time a single logic output. Since the output is also a logic-level value, an output of one logic gate can connect to the input of one or more other logic gates. The logic thereby performed is thus adequate for the functioning of digital circuits. Logic gates are typically implemented electronically using diodes or transistors but, as discussed below, can they also be constructed using inter alia promoters and regulators. An archetypical example in this context is the *lac* operon of *E. coli*, where expression of the genes for lactose metabolism is controlled by the lacI repressor and by the cyclic AMP receptor protein (CRP) activator. The LacI repressor binds to the *lac* promoter (P_{lac}) as a tetramer and inhibits gene expression both through the physical occupation of the RNAP binding site and through the formation of a DNA loop [12]. The binding of the inducer (lactose or IPTG) to LacI triggers a conformational switch in the tetramer that decreases the affinity to the operator sequences and thus allows transcription initiation from the P_{lac} [12,13]. The behaviour of the *lac* regulatory system has been described to be an intermediate between AND-gate and OR-gate logic function (see below; [14]).

Although binary logic circuits are based on functions with just two possible states (0 or 1), existing biological systems typically display continuous values for the input/output functions [15]. In addition, such values are submitted to noise and cell-to-cell stochastic variations due to the nature of the molecular interactions involved [16]. This has important consequences for the construction of artificial genetic circuits based in the naturally occurring transcriptional modules and its applicability in synthetic networks [17]. For example, an artificial system with oscillatory properties constructed by the combination of the repressor properties of three well characterized TFs (LacI, TetR and the λ repressor), lost its periodicity after a few rounds of oscillation [18]. Although promoters destined for building artificial circuits should ideally behave as bi-stable switches resembling a digital response, this is not the case in most available instances. Whether or not naturally occurring promoters can be artificially re-designed to achieve permanently such a binary performance remains an open question, as Darwinian selection may eventually press against such a conduct.

2.2. Simple logic gates shape the bulk of transcriptional regulation circuits

Despite the constraints mentioned above, representing the reactions and interactions involved in gene expression control using circuit diagrams and Boolean logic operators is still an useful abstraction. As the biological reactions adopt somewhat continuous values, the 0/1 states are generally agreed to reflect low/high states for the *input* status and off/on for *output* promoter activity. Figs. 1 and 2 summarize the most relevant logic gates that have been either described experimentally or suggested to occur on the basis of simulations using empirical data. The schemes of Figs. 1 and 2 do not cover all possible combinations of regulatory modules that can originate the diverse gates shown, but they illustrate each case with a simplified biological example.

The two simplest logic gates that describe biological functions include one promoter regulated by one activator or by one repressor. In the first case we have the so-called *buffer-gate*

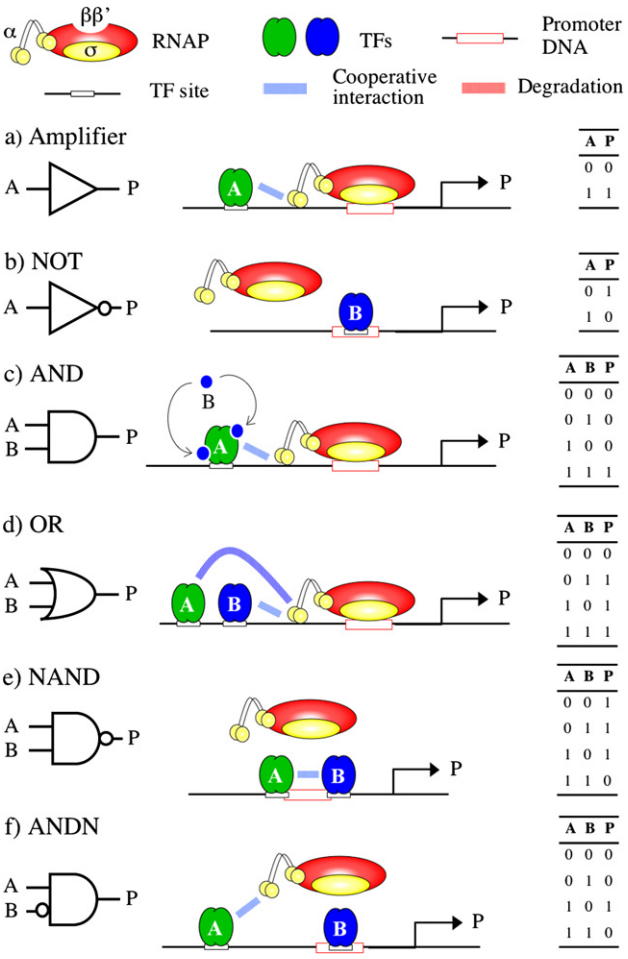

Fig. 1. Models of logic gates built with prokaryotic regulatory modules. The sketches on top of the figure symbolize the various actors that control promoter activity: RNA polymerase (RNAP) disclosed in its various subunits, transcription factors (activators and/or repressors), binding DNA sites and types of interaction. Each of the gate models are assembled by combination of TFs binding sites (operators) and RNAP binding sites (promoters). The overlapping of one operator with a promoter causes repression while operators placed upstream from the promoter causes activation. (a) The amplifier-gate is represented as a simple activation process. (b) The NOT-gate is equivalent to transcriptional repression. (c) The AND-gate can be implemented as a TF that depends on an inducer B to activate the promoter. (d) The OR-gate could be a promoter amenable to full activation by two independent TFs. (e) One NAND-gate is generated by a promoter regulated by two cooperative repressors. (f) An ANDN-gate can be created with a promoter activated by a TF and repressed by another.

or *amplifier-gate*, where the output has the same state that the input (i.e., if the input is low the output is *off* and vice-versa, Fig. 1A). For a repressor, the system is represented as a NOT-gate, where the promoter is active (*on*) only in the absence of the repressor (the *low* state, Fig. 1B). The graphical difference between these two gates is the presence of an inverting *bubble* on the output terminal of the NOT-gate.

For the systems where two inputs are computed to generate one output, there are 16 possibilities of Boolean logic gates (2^n, where $n = 4$ combinations of input states) [19]. However, there seems to be only eight biologically relevant gates, as analyzed previously [2]. The AND-gate represents a regulatory system