Project no. 043338

Project acronym: EMERGENCE

Project title: A foundation for Synthetic Biology in Europe

Instrument: NEST Pathfinder

Thematic Priority: Synthetic Biology

D3.3: Report describing the software for model-based systems design and analysis, and its integration

Due date of deliverable: M24
Actual submission date: M24

Start date of project: 1.12.2006 Duration: 36 months

Organisation name of lead contractor for this deliverable: ETHZ

Version 1.0

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Objectives and Background

The work associated with the deliverable aims at the development of a software tool for the model-based design and analysis of synthetic gene circuits, and the integration of this software with other components of the IT infrastructure such that models and components are re-usable, standardized, and that they include references to standard building blocks for synthetic biology ('BioBricks'). In addition, the software tools need to be user-friendly (e.g. by providing graphical user interfaces for circuit specification and analysis) to make them practically available to the interdisciplinary community of synthetic biology researchers. Corresponding software tools are currently lacking, which impedes the progress in rational design of biological systems, which constitutes a bottleneck for the development of a European IT Infrastructure for synthetic biology.

Deliverable Procedure

The work started from the principle that novel genetic circuits can be engineered using standard parts with well-understood functionalities. However, no model based on the simple composition of these parts has become a standard, mainly because it is difficult to define signal exchanges between biological units as unambiguously as in electrical engineering. Taking inspiration from (and slightly modifying) ideas in the 'MIT Registry of Standard Biological Parts', we developed a method for the design of genetic circuits with composable parts. Conceptual extensions concern the fact that gene expression requires four kinds of signal carriers: RNA polymerases, ribosomes, transcription factors and environmental 'messages' (inducers or corepressors). The flux of each of these types of molecules is a quantifiable biological signal exchanged between parts. To achieve an implementation of the concept, each part is modeled independently by the ordinary differential equations (ODE) formalism. The implementation relies on the integration of derived (modular) models with the established software tool ProMoT (Process Modeling Tool). This allowed to realize a 'drag and drop' tool, where genetic circuits are built just by placing biological parts on a canvas and by connecting them through 'wires' that enable flow of signal carriers, as it happens in electrical engineering. Models can then be exported in a standard exchange format (SBML) for integration with other software components such as simulation or analysis platforms. As a first proof-of-principle, simulations of well-known synthetic circuits agree well with published computational and experimental results.

Availability

Detailed methods and descriptions of the software tool were published (see Appendix I). Software implementations are available from partner ETHZ and it is planned to enable broader distribution by establishing a dedicated web site.

Appendix I

Reprint of related publication:

Marchisio, M.A. and J. Stelling, *Computational design of synthetic gene circuits with composable parts.* Bioinformatics, 2008. **24**(17): p. 1903-10.

BIOINFORMATICS ORIGINAL PAPER Vol. 24 no. 17 2008, pages 1903–1910
doi:10.1093/bioinformatics/btn330

Systems biology

Computational design of synthetic gene circuits with composable parts

M.A. Marchisio* and J. Stelling

Department of Biosystems Science and Engineering and Swiss Institute of Bioinformatics, ETH Zurich, 8092 Zurich, Switzerland

Received on February 7, 2008; revised on May 28, 2008; accepted on June 23, 2008

Advance Access publication June 25, 2008

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: In principle, novel genetic circuits can be engineered using standard parts with well-understood functionalities. However, no model based on the simple composition of these parts has become a standard, mainly because it is difficult to define signal exchanges between biological units as unambiguously as in electrical engineering. Corresponding concepts and computational tools for easy circuit design in biology are missing.

Results: Taking inspiration from (and slightly modifying) ideas in the 'MIT Registry of Standard Biological Parts', we developed a method for the design of genetic circuits with composable parts. Gene expression requires four kinds of signal carriers: RNA polymerases, ribosomes, transcription factors and environmental 'messages' (inducers or corepressors). The flux of each of these types of molecules is a quantifiable biological signal exchanged between parts. Here, each part is modeled independently by the ordinary differential equations (ODE) formalism and integrated into the software ProMoT (Process Modeling Tool). In this way, we realized a 'drag and drop' tool, where genetic circuits are built just by placing biological parts on a canvas and by connecting them through 'wires' that enable flow of signal carriers, as it happens in electrical engineering. Our simulations of well-known synthetic circuits agree well with published computational and experimental results.

Availability: The code is available on request from the authors.

Contact: mario.marchisio@bsse.ethz.ch

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Synthetic biology deals with the purpose-driven design and implementation of novel biological functions such as engineered genetic circuits. The field has spurred the interest of many research groups that made efforts to build biological devices by means of well-known genetic structures. Application areas can be found in fields from environmental sciences to medicine and diagnostics (Sayut *et al.*, 2007) and several remarkable engineered biological circuits have been realized (for reviews see, for instance, Andrianantoandro *et al.*, 2006; Benner and Sismour, 2005; Drubin *et al.*, 2007; Hasty *et al.*, 2002).

In general, biological circuits can be constructed from a handful of basic parts. To completely implement a (basic) transcription unit, for instance, one needs promoters, ribosome binding sites (RBS), protein coding regions and terminators. Other parts encoding for spacers or for particular stem-loop RNAs can fine-tune gene regulation, or allow more degrees of freedom in controlling gene expression. An exhaustive repository of synthetic parts is the 'MIT Registry of Standard Biological Parts' (http://partsregistry.org/), a reference point for current research in synthetic biology. It contains not only basic parts but also more complex devices accompanied by some relevant information about their structures and functions. However, to build devices from basic parts efficiently, the complexity of the reactions as well as the variety of the molecules involved make it very difficult to accurately predict the behavior even of simpler biological devices.

For transcription networks, nevertheless, a qualitative depiction of their response to stimuli and an estimation of produced proteins can be obtained by employing mathematical modeling frameworks such as the ordinary differential equation (ODE) formalism (Alon, 2006). In a rough approximation, mRNA transcription and translation are treated as a single step. Control of gene expression—which may involve cooperativity, competition between transcription factors and processing of environmental signals—can be described by an appropriate choice of Michaelis–Menten type reaction kinetics and coefficients. Protein production then depends on the activity of the corresponding transcription units, the translation rates and the proteins' (constant) decay rates.

A more detailed view, which allows an estimation of the time delay between transcription and translation, demands to separate these two events by explicitly modeling the mRNA dynamics (Klipp *et al.*, 2005). This more accurate description of the system dynamics increases the number of model parameters. As many of the associated kinetic parameters have not been unequivocally determined yet, this adds uncertainty to the prediction of the system behavior (Tomshine and Kaznessis, 2006). A more realistic insight into a biological network can be obtained by treating it as a stochastic system. However, under precise conditions (as stated in Samoilov and Arkin, 2006) the ODE formalism is the continuous–deterministic limit of a discrete–stochastic system description. Hence, trade-offs between model accuracy and efforts needed for establishing the model are important considerations for synthetic biology, and generalizable frameworks are needed.

*To whom correspondence should be addressed.

Moreover, independent of the representation, a mathematical model of a biological circuit can hardly be based directly on the Registry's basic parts. Currently, the parts are not *composable*, that is, they do not share the same types of input and output. In circuit design for electrical engineering, parts such as batteries, resistors and solenoids can be assembled in many different ways because they all exchange information via the common 'currency' of a flux of charged particles that can be measured easily. This suggests that the implementation of biological circuits requires an exchange of information by fluxes of common signal carriers as well. Such a framework would enable us to represent biological networks more intuitively by separated modules (the parts) connected by wires. However, there exists no commonly accepted biological counterpart of the electric current yet. Mathematical models of genetic circuits based on the Registry parts are, in general, treated as unique structures that show no modularity. Hence, we urgently need concepts and tools for systematic computational design from reusable parts as in other engineering disciplines as well as a database of the Registry part models, as pointed out in Rouilly *et al.* (2007).

A corresponding concept can start from the realization that the expression of every gene needs RNA polymerases (for transcription) and ribosomes (for translation). The 'Abstraction Hierarchy' pages of the Registry propose the flux of these signal carriers as units for characterizing the information exchange between parts. Polymerases per second (*PoPS*) and ribosomes per second (*RiPS*) could allow parts to communicate to each other just by means of a 'current' of polymerases and ribosomes (Endy *et al.*, 2005). This picture, however, does not seem sufficient to describe all information exchanges even in simple engineered gene circuits. We argue that other signal carriers like transcription factors and environmental 'messages' should be explicitly introduced and not indirectly estimated by means of *PoPS* and *RiPS*. This permits modeling the reactions involved in protein synthesis more precisely without loss of parts composability. Furthermore, we introduce *pools* of proteins and small molecules. They are connected to every transcription unit and distribute 'free' signal carriers correctly among the parts according to their affinities. These pools allow *scalability*; the system response to different signal carrier concentrations is particularly important in complex network simulations as shown in Supplementary Material.

Several software tools have functionalities analogous to those for electrical circuit design, but none of them combines ease-of-use, parts composability, and detailed modular modeling approaches. BioJADE (Goler, 2004) as one of the first tools provides a graphical user interface (GUI) to place, connect and even modify Registry parts, but it considers only one kind of signal carrier (RNA polymerases) and, hence, very simplified models of gene expression dynamics. CellDesigner (Funahashi *et al.*, 2003) has similar capabilities for graphical circuit composition. However, parts modularity and, consequently, circuit representation do not appear detailed enough because the Hill functions (Kærn and Weiss, 2006) employed assume *quasi-equilibrium* conditions. Total RNA polymerase and ribosome concentrations are *de facto* ignored and signal carriers are absent—this prevents precise simulations of large engineered networks. A very recent tool, Asmparts (Rodrigo, G. *et al.*, submitted for publication in *Systems and Synthetic Biology*), applies the same Hill formalism for the Registry parts, providing SBML code for each of them. Parts can be assembled from the command line (but not a GUI) into a unique circuit file. *PoPS* and *RiPS* based on the Hill functions are formally derived, but they are not explicitly computed. Transcription factors (but not their fluxes, or environmental signals) are included as promoter input; however, we think that it is necessary to model each part in more detail to better depict the signal carrier dynamics (see Section 3). An opposite approach is realized in TABASCO (Kosuri *et al.*, 2007), which emphasizes the action of RNA polymerases and ribosomes at single base-pair resolution. The tool permits to estimate gene expression with high precision and it is a powerful instrument for circuit simulations. Nevertheless, it lacks part modularity, which limits the use for circuit design.

Here, we present a new framework for the design of synthetic circuits where each part is modeled independently following the ODE formalism. This results in a set of composable parts that communicate by fluxes of signal carriers, whose overall amount is constantly updated inside their corresponding pools. Basic parts, moreover, can be put together to build *composite devices* such as protein generators, reporters and inverters. Again, these are composable and able to communicate both with parts and pools. We have implemented the corresponding models into ProMoT (Process Modeling Tool), a software for the object-oriented and modular composition of models for dynamic processes (Ginkel *et al.*, 2003). This tool allows one to design a synthetic biological circuit easily, just by displaying its parts on the screen and by connecting them by 'wires' for the signal carrier exchange. We test the concept by representing some of the most well-known synthetic circuits: both their qualitative and quantitative behaviors can be fairly reproduced.

2 APPROACH

For modeling general genetic circuits, we can start by considering a simple one-step cascade circuit (Fig. 1). This small network needs at least four different kinds of signal carriers, namely RNA polymerases, ribosomes, transcription factors and environmental signals. To each of these (classes of) molecules, we can associate a different unit to quantify its flux along the parts: the already mentioned *PoPS* and *RiPS* as well as the factor per second (*FaPS*) and the Signal Per Second (*SiPS*). Following the Registry, *PoPS* can be defined as the quantity of RNA polymerases that passes a defined point on the DNA per time with unit molars per second (M/s). An analogous definition is valid for *RiPS*. *FaPS* are the quantity of transcription factors (activators or repressors) produced per second inside their corresponding coding regions. *SiPS* represent the amount of environmental signals (inducers or corepressors) that enters the cell per time unit. Thus, every flux is just a derivative of a concentration with respect to time so that it is straightforward to integrate it into an ODE-based model.

Every part is, hence, able to calculate one or more of these basic fluxes and to communicate them to the connected parts whose functioning is affected by this information. Note that composable parts do not need to exchange all four types of molecules, but just the ones they are interested in. In other words, parts composability does not mean that the parts themselves can be put together randomly inside a circuit, but they have to obey some biological constraints. For instance, a functional protein coding region cannot be connected directly to a promoter because it has to be preceded by a ribosome binding site to be translated. The composition of a synthetic circuit can be validated with parsing algorithms (Cai *et al.*, 2007). Furthermore, the total quantities of free signal carriers have to be updated continuously and must be visible to every

interested part. Hence, every promoter inside a circuit has to be connected to a polymerase pool. Additional connections to one or more signal and transcription factor pools depend on the nature of the promoter. Analogously, ribosome binding sites as well as protein coding regions must be connected to the ribosome pool. The coding regions, furthermore, access transcription factor pools whenever transcriptional repressors or activators are their products. Terminators, on the contrary, interact just with the polymerase pool, sending a flux of molecules that have finished the transcription of a gene. This picture implies that, for instance, the promoter is not a simple *PoPS* 'battery' that creates a signal *de novo*. The signal produced inside a promoter is regulated by the total pool of free polymerases and by the action of transcription factors, inducers and corepressors. All promoters constantly exchange information with the polymerase pool, leading to an interconnected network of genes.

The above units that characterize the exchange of signal carriers between parts are difficult to measure experimentally, for instance, because the molecules move discontinuously along a nucleotide chain or inside the cell. In our view, the strength of the concept is not to try to estimate the behavior of a given device just in terms of *PoPS* as inputs and outputs. Common signal carrier fluxes are most useful in providing abstractions that make parts composable and, consequently, facilitate design and simulation of biological circuits. The circuits' behavior will still be described in terms of protein produced per time or as a function of inducer/corepressor concentrations, for instance. Note that different networks might require other basic parts, which can imply the construction of new pools and the exchange of other signal carriers. This applies, for instance, to non-coding RNA parts (see Supplementary Material).

3 METHODS

Even though, like in the most traditional approach, we use the ODE formalism, the novelty of our method lies in the *composability* of parts. The parts are modeled independently and can be interconnected through the exchange of common signal carriers whose fluxes are expressed in the units explained in the previous section. In the following, we describe in detail the parts necessary to build a one-step cascade (Fig. 1). All variables represent concentrations (in M) except for the fluxes. Quantities in square brackets refer to biochemical complexes.

3.1 The *basic* promoter

The first transcription unit of a one-step-cascade network encodes for a transcription factor, namely a repressor. Its expression is supposed to be independent of any other transcription factors in the cell, so that it can be estimated by using an (unrealistic) *basic* promoter without operators. The promoter interacts just with RNA polymerases. We assume an initial condition where all the RNA polymerases are free (Pol^{free}) and stored inside their pool. They are seen by free promoters (P) and can bind to them following a Michaelis–Menten schema

$$Pol^{free} + P \overset{(k_1,k_{-1})}{\rightleftharpoons} [PPol] \xrightarrow{k_2} P + Pol^{cl} \tag{1}$$

where $[PPol]$ represents the *initiation* complex formed by a polymerase and a promoter; Pol^{cl} refers to the RNA polymerase in the *clearance* phase during which transcription initiation is completed. The kinetic constants k_1 and k_{-1} are related to the formation and the dissociation of the $[PPol]$ complex, whereas k_2 is the transcription initiation frequency.

As the total promoter concentration (P_T) is fixed and given by the sum of free and occupied promoters: $P_T = P + [PPol]$, the state of the promoter

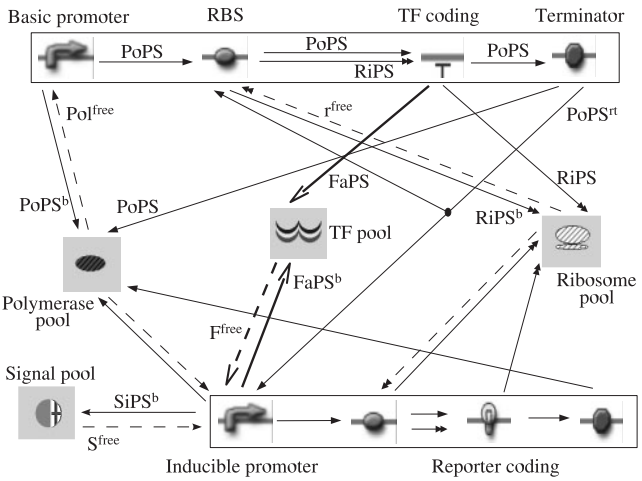


Fig. 1. One-step cascade network. The first transcription unit (box on the top) encodes for a repressor for the promoter leading the second transcription unit (box on the bottom) that produces a reporter protein. Environmental signals entering the inducible promoter can inactivate repressors and turn on protein synthesis. Solid and dashed arrows represent the fluxes and the available concentrations of the four different signal carriers, respectively (simple arrows: *PoPS* and Pol^{free}; double arrows: *RiPS* and r^{free}; line arrows: *FaPS* and F^{free}; concave arrows: *SiPS* and S^{free}). The transcription units are associated with two different composite devices: a *protein generator* and a *reporter*.

is captured by the $[PPol]$ amount, which follows the differential equation

$$\frac{\mathrm{d}[PPol]}{\mathrm{d}t} = k_1 Pol^{free} P_T - (k_1 Pol^{free} + k_{-1} + k_2)[PPol]. \tag{2}$$

Two different polymerase fluxes leave the promoter part: one is a negative 'balance' flux ($PoPS^b$) sent to the polymerase pool, which corresponds to the variation of free polymerase concentrations due to the interaction with the promoter

$$PoPS^b = -k_1 Pol^{free} P_T + (k_1 Pol^{free} + k_{-1})[PPol] \tag{3}$$

and the other is the outgoing flux ($PoPS^{out}$) directed to the next part in the transcription unit (in this case an RBS)

$$PoPS^{out} = k_2 [PPol]. \tag{4}$$

From Equation (1), it is apparent that $PoPS^{out}$ is nothing else than the time derivative of the polymerase concentration in the clearance phase, Pol^{cl}.

3.2 The RBS

The polymerases per second leaving the promoter [see Equation (4)] represent the input signal for the RBS ($PoPS^{in}$). All the incoming RNA polymerases are supposed to bind, at the beginning of this region, to a site that we will call B. This gives rise to a new complex ($[PolB]$) before starting *mRNA* transcription with a constant elongation velocity:

$$PoPS^{in} \Longrightarrow [PolB] \xrightarrow{k_{el}^{RBS}} Pol^{el} + B. \tag{5}$$

Note that, in principle, this model does not force RNA polymerase to have the same velocity inside different parts. The $[PolB]$ complex is an artifact to model passage of the RNA polymerases from the clearance to the elongation phase (Pol^{el}). The rate of Pol^{el} formation (k_{el}^{RBS}) is given by the ratio of the elongation velocity (v_{el}) to the RBS length (l_{RBS}): $k_{el}^{RBS} = v_{el}/l_{RBS}$.

Equation (5) allows us to estimate the amount of $PoPS^{out}$ leaving the RBS part. It is the time derivative of Pol^{el}, which corresponds to

$$PoPS^{out} = k_{el}^{RBS} [PolB], \tag{6}$$

so that the time derivative of $[PolB]$ corresponds to the algebraic sum of the incoming and the outgoing polymerase fluxes

$$\frac{\mathrm{d}[PolB]}{\mathrm{d}t} = PoPS^{in} - PoPS^{out}. \tag{7}$$

While the mRNA leader region is transcribed, we assume that free ribosomes (r^{free}) leave their pool and bind to a binding site (b) on the mRNA forming the $[rb]$ complex with Michaelis–Menten type kinetics

$$r^{free} + b \overset{(k_{1r}, k_{-1r})}{\rightleftharpoons} [rb] \overset{k_{2r}}{\rightarrow} r^{cl} + b, \tag{8}$$

where r^{cl} represents the ribosome concentration during the leader clearance phase. Just after clearing the RBS completely, ribosomes can bind to the start codon (AUG) located in the next part, the protein coding region, and start protein synthesis. Note that, similar to the $[PolB]$ complex, the $[rb]$ complex does not really exist. Ribosomes bind directly to the AUG codon, which belongs to the RBS. Nevertheless, the $[rb]$ complex is instrumental in estimating the initial value of *RiPS* directly from the translation initiation frequency (k_{2r}), which corresponds to the inverse of the RBS clearance time. Whereas the promoter generates a *PoPS* signal, the RBS is the *RiPS* signal generator.

From Equation (8), we can derive the time derivative of r^{cl}, which corresponds to the ribosome flux that leaves the RBS

$$RiPS^{out} = k_{2r}[rb]. \tag{9}$$

The *free* mRNA concentration (b) depends on the polymerase flux, the interaction with ribosomes [Equation (8)], and the mRNA degradation constant (k_d):

$$\frac{\mathrm{d}b}{\mathrm{d}t} = k_{el}^{RBS}[PolB] - k_{1r} r^{free} b + (k_{-1r} + k_{2r})[rb] - k_d b + PoPS^{rt}. \tag{10}$$

Here, furthermore, we included a term due to transcriptional *readthrough* ($PoPS^{rt}$), that is, the polymerase flux that passes the terminator and enters the next promoter. Assuming that $[rb]$ decays with the same degradation constant as b (producing free ribosomes), the time dependency of the $[rb]$ complex concentration obeys the differential equation

$$\frac{\mathrm{d}[rb]}{\mathrm{d}t} = k_{1r} r^{free} b - (k_{-1r} + k_{2r} + k_d)[rb] \tag{11}$$

and the ribosomes per second exchanged with the ribosome pool ($RiPS^b$) are given by

$$RiPS^b = -k_{1r} r^{free} b + (k_{-1r} + k_d)[rb]. \tag{12}$$

Note that $RiPS^b$ is a negative flux of ribosomes directed from the RBS to the ribosome pool.

Hence, the RBS part handles two different signal carriers: RNA polymerases and ribosomes. This permits to completely evaluate the total mRNA concentration in the system [Equation (10)], although the transcription process continues in the protein coding part, just by extending the mRNA chains here initiated.

3.3 The protein coding part

Both the polymerase and the ribosome flux produced inside the RBS go into a protein coding part representing a gene. Incoming RNA polymerases are supposed to form a new complex by binding to the *start point* position on the DNA (A) before going on transcribing the mRNA with the same average elongation velocity as inside the RBS

$$PoPS^{in} \Longrightarrow [PolA] \overset{k_{el}^{PC}}{\rightarrow} Pol^{el} + A. \tag{13}$$

Macroscopically, the transcription rate k_{el}^{PC} is much smaller than inside the RBS (k_{el}^{RBS}) because it is inversely proportional to the length of the gene. As for the RBS, the outgoing polymerase flux, directed this time to a terminator, is given by the expression

$$PoPS^{out} = k_{el}^{PC}[PolA] \tag{14}$$

and the time derivative of the $[PolA]$ complex follows the equation

$$\frac{\mathrm{d}[PolA]}{\mathrm{d}t} = PoPS^{in} - PoPS^{out}. \tag{15}$$

A flux of ribosomes also enters this part. Ribosomes bind to the mRNA at the start codon (AUG), forming a complex indicated as $[ra]$. They translate mRNA until they encounter the stop codon (XXU), bind to it, and form another complex, $[ru]$. At this point, ribosomes are freed again and go back to their pool. Hence, whereas we have, as inside the other parts, just two polymerase fluxes ($PoPS^{in}$ and $PoPS^{out}$), one more flux is required to describe the ribosome dynamics. It is associated with the internal flux of ribosomes between the complexes $[ra]$ and $[ru]$ ($RiPS^{PC}$)

$$RiPS^{in} \Longrightarrow [ra] \overset{k_{el}^{r}}{\Longrightarrow} RiPS^{PC}, \tag{16}$$

$$RiPS^{PC} \Longrightarrow [ru] \overset{\zeta_r}{\rightarrow} r^{free} + XXU. \tag{17}$$

The ribosome elongation rate (k_{el}^{r}) is the ratio of the average translational elongation velocity (v_{el}^{r}) to the gene length. From Equation (16), we have

$$RiPS^{PC} = k_{el}^{r}[ra] \tag{18}$$

whereas the outgoing flux of free ribosomes toward their pool can be obtained from Equation (17)

$$RiPS^{out} = \zeta_r[ru]. \tag{19}$$

Here, ζ_r is the ribosome dissociation constant; it depends on the particular release factors involved in the translation termination process. The variation of $[ra]$ and $[ru]$ with respect to time is given by

$$\frac{\mathrm{d}[ra]}{\mathrm{d}t} = RiPS^{in} - RiPS^{PC} \text{ and} \tag{20}$$

$$\frac{\mathrm{d}[ru]}{\mathrm{d}t} = RiPS^{PC} - RiPS^{out} \tag{21}$$

whereas the total amount of synthesized protein (z) can be obtained by

$$\frac{\mathrm{d}z}{\mathrm{d}t} = RiPS^{PC} - k_D z \tag{22}$$

with the protein decay constant k_D. When z is a repressor or an activator, the coding part communicates with the appropriate transcription factor pool by a flux of proteins ($FaPS^{out}$):

$$FaPS^{out} = RiPS^{PC}. \tag{23}$$

Note that Equation (23) has no degradation term because it is calculated only once inside the pool.

3.4 The terminator

The RNA polymerases leaving the protein coding region enter the terminator (T) where they form a new complex ($[PolT]$) before becoming free and flowing again to their pool ($PoPS^{out}$):

$$PoPS^{in} \Longrightarrow [PolT], \tag{24}$$

$$[PolT] \overset{\zeta}{\longrightarrow} \tag{25}$$

$$Pol^{free} + T, PoPS^{out} = \zeta[PolT]. \tag{26}$$

Depending on the terminator's efficiency (e), however, a fraction of the polymerases engaged in the $[PolT]$ complex may continue processing the next transcription unit. This generates a readthrough flux ($PoPS^{rt}$)

$$[PolT] \overset{\eta}{\longrightarrow} Pol^{rt} + T, \tag{27}$$

$$PoPS^{rt} = \eta[PolT]. \tag{28}$$

The dissociation constant ζ and the readthrough constant η provide the terminator efficiency as: $e = \zeta/(\zeta + \eta)$. The time derivative of the $[PolT]$

complex corresponds to the sum of the incoming and outgoing polymerase fluxes

$$\frac{\mathrm{d}[PolT]}{\mathrm{d}t} = PoPS^{in} - PoPS^{out} - PoPS^{rt}. \quad (29)$$

This part usually terminates a transcription unit, although in many cases it can be followed by another terminator to reduce the readthrough effect.

3.5 The one-operator promoter

In the single-step cascade, the reporter's transcription unit is lead by an inducible promoter with one operator that can host repressors from the transcription factor pool. Inducers from the signal pool can enter the promoter part, bind to the repressors, and inactivate them. This increases the probability that RNA polymerases transcribe the reporter. Instead of the variable P used for the basic promoter, it is convenient to introduce a new variable O for the operator state. It can take two values: *free* (O^f), available to the RNA polymerase and *taken* (O^t), occupied by a repressor.

The Michaelis–Menten reaction of Equation (1) then becomes

$$Pol^{free} + O^f \overset{(k_1,k_{-1})}{\rightleftharpoons} [PolO^f] \xrightarrow{k_2} O^f + Pol^{cl}. \quad (30)$$

Repressors arrive at the promoter in their active form (R^a) and can interact directly with free operators and inducers (I)

$$R^a + O^f \overset{(\alpha,\beta)}{\rightleftharpoons} O^t \;;\; nI + R^a \overset{(\lambda,\mu)}{\rightleftharpoons} R^i \quad (31)$$

where n is the number of inducer molecules that cooperatively turn an active repressor into the inactive form (R^i). The value of n is lower than or equal to the number of repressor subunits. Inducers can also release repressors bound to the operators

$$nI + O^t \xrightarrow{\gamma} R^i + O^f. \quad (32)$$

We assume that repressors always decay with rate constant k_D, independent of their binding state. However, inside the promoter we calculate only R^i and O^t degradation explicitly, whereas R^a are supposed to decay inside their pool. Hence, we have

$$R^i \xrightarrow{k_D} nI \;;\; O^t \xrightarrow{k_D} O^f. \quad (33)$$

This promoter handles three different signal carriers: transcription factors (repressors), environmental signals (inducers) and RNA polymerases. It exchanges up to five different fluxes of these molecules. The fluxes of transcription factors and environmental signals are negative and directed toward the corresponding pools. $FaPS^b$ represents the time derivative of the free active repressor (R^a) due to the reactions in Equation (31)

$$FaPS^b = -\alpha R^a O^f + \beta O^t - \lambda R^a I^n + \mu R^i, \quad (34)$$

whereas $SiPS^b$ reflects the time variation of the total free inducer concentration caused by Equations (31–33)

$$SiPS^b = n(-\lambda R^a I^n + \mu R^i - \gamma O^t I^n + k_D R^i). \quad (35)$$

Note that the free promoter/operator concentration O^f in Equation (34) can be derived from: $O_T = O^f + [PolO^f] + O^t$, where O_T is the total promoter concentration. RNA polymerase is involved in four different fluxes, three of them exchanged with external parts. The promoter is connected to the terminator of the first transcription unit, from which it receives a readthrough signal ($PoPS^{rt}$). We can assume that these polymerases encounter only free promoters, which results in an increment of $[PolO^f]$

$$PoPS^{rt} \Longrightarrow [PolO^f]. \quad (36)$$

Furthermore, RNA polymerases can bind weakly to occupied promoters O^t, yielding a *leakage* flux ($PoPS^{lk}$) according to:

$$PoPS^{lk} = k_2^{lk} O^t \quad (37)$$

where the *basal* transcription initiation frequency k_2^{lk} is generally much lower than k_2. Leakage contributes to the outgoing polymerase flux and to the negative flux back to the polymerase pool, respectively:

$$PoPS^{out} = k_2[PPol] + PoPS^{lk} \quad (38)$$

$$PoPS^b = -k_1 Pol^{free} O^f + k_{-1}[PPol] - PoPS^{lk}. \quad (39)$$

Conversely, polymerase readthrough enters the $[PolO^f]$ state equation

$$\frac{\mathrm{d}[PolO^f]}{\mathrm{d}t} = k_1 Pol^{free} O^f - (k_{-1} + k_2)[PolO^f] + PoPS^{rt} \quad (40)$$

[compare to the basic promoter, Equation (2)]. To complete the one-operator promoter description, we need two more ODEs for O^t and R^i, respectively:

$$\frac{\mathrm{d}O^t}{\mathrm{d}t} = \alpha R^a O^f - (\beta + \gamma I^n + k_D) O^t, \quad (41)$$

$$\frac{\mathrm{d}R^i}{\mathrm{d}t} = \lambda R^a I^n - \mu R^i + \gamma O^t I^n - k_D R^i. \quad (42)$$

3.6 The signal carrier pools

All pools in our model are new parts and not yet included in the Registry. The polymerase pool stores all free RNA polymerase molecules; it is connected to every promoter and terminator in a circuit. The total amount of free polymerases, constantly visible to the promoters, is calculated by the negative $PoPS^b$ flux from the promoter parts [Equations (3, 39)] and the $PoPS^{in}$ flux from the terminator parts [Equation (26)]

$$\frac{\mathrm{d}Pol^{free}}{\mathrm{d}t} = \sum_{i=1}^{N} (PoPS_i^b + PoPS_i^{in}), \quad (43)$$

where N is the number of transcription units in the network. The ribosome pool functions identically, but it is connected to the RBS and the protein coding parts. Hence, the free ribosome concentration is given by

$$\frac{\mathrm{d}r^{free}}{\mathrm{d}t} = \sum_{i=1}^{N} (RiPS_i^b + RiPS_i^{in}) \quad (44)$$

where $RiPS^b$ is a negative flux [calculated in Equation (12)] and $RiPS^{in}$ coincides with the quantity in Equation (19). The RBS part has constant access to the value of r^{free} updated through Equation (44).

In our example network, repressors are produced by the transcription factor coding part of the first gene. Repressor monomers (F^m) are sent to the transcription factor pool [Equation (23)]

$$FaPS^{in} \Longrightarrow F^m, \quad (45)$$

where they may dimerize (or: form higher order complexes) to enable interactions with operators and inducers

$$2F^m \overset{(\delta,\epsilon)}{\rightleftharpoons} F^{free}. \quad (46)$$

Here, δ and ϵ are the complex association and dissociation rate constants, respectively. Free dimers (F^{free}) coincide with the active repressors (R^a) that regulate the one-operator promoter. This results in a 'balance', negative $FaPS^b$ flux in Equation (34) from the promoter to the pool

$$FaPS^b \Longrightarrow F^{free}. \quad (47)$$

Free dimers and monomers are supposed to decay with identical rates (k_D)

$$F^m \xrightarrow{k_D} ; F^{free} \xrightarrow{k_D}. \quad (48)$$

Again, the role of the transcription factor pool is to update the total concentrations of free, active transcription factors by

$$\frac{\mathrm{d}F^{free}}{\mathrm{d}t} = \delta F^{m2} - \epsilon F^{free} - k_D F^{free} + FaPS^b \quad (49)$$

where F^m obeys the following differential equation

$$\frac{\mathrm{d}F^m}{\mathrm{d}t} = 2(-\delta F^{m2} + \epsilon F^{free}) - k_D F^m + FaPS^{in}. \quad (50)$$

Furthermore, as mentioned above, we use a model structure where free factors decay inside the pool, whereas factors bound to n signals are degraded inside the promoter part.

For free signals (inducers, S^{free}), we assume constant production

$$\xrightarrow{k} S^{free} \tag{51}$$

in their pool with production rate constant k. Free inducers can bind to the promoter and deactivate repressors as stated in Equations (31, 32). This creates a negative flux ($SiPS^b$) from the promoter to the pool [Equation (35)]

$$SiPS^b \Longrightarrow S^{free}. \tag{52}$$

Free signal degradation takes place inside the pool, with a decay rate (k_{D_s}) that is small compared to the one of the associated transcription factor

$$S^{free} \xrightarrow{k_{D_s}}. \tag{53}$$

Clearly, the signal pool is needed to calculate the total concentration of free signal at each time step and to communicate it to the connected promoter(s)

$$\frac{\mathrm{d}S^{free}}{\mathrm{d}t} = k - k_{D_s} S^{free} + SiPS^b. \tag{54}$$

4 IMPLEMENTATION

As briefly mentioned in Section 1, ProMoT is a systems modeling and design tool that permits to reproduce the dynamics of a biochemical system through *modules*. Each module represents a system subunit, characterized by a certain degree of complexity and autonomy. It can estimate the temporal evolution of some general quantities and communicate it to other modules. Each of our biological parts (the basic ones as well as the composite devices) is associated with an appropriate module. Therefore, we encoded each part in MDL (Modeling Description Language), the object-oriented Lisp-based programming language of ProMoT. ProMoT, furthermore, provides the user with a Java GUI where one can just drag and drop the parts needed, without caring of their content, and then connect them through 'wires', as it is done in many electrical engineering tools [see, for instance, SPICE (Nagel and Pederson, 1973)].

More specifically, the MDL code of the parts needs the definition of *variables*, *terminals* and *equations*. Variables represent all time-varying quantities (state variables for ODE systems) as well as the constant parameters. Terminals are the interfaces between parts and contain all the variables necessary for information exchange. Equations can be simple algebraic relations or ODEs. The one-operator promoter for instance, has five terminals. One terminal connects to a terminal of the polymerase pool to get the amount of available free RNA polymerases (Pol^{free}) and to communicate the value of $PoPS^b$. A second terminal sends the produced $PoPS^{out}$ to another part (an RBS for instance). The last terminal associated with RNA polymerase will receive $PoPS^{rt}$ from an adjacent terminator. Two more terminals connect the promoter to the transcription factor and to the signal pool. These terminals receive the total amounts of free molecules (F^{free} and S^{free}) and send the values of $FaPS^b$ and $SiPS^b$, respectively. Note that whenever a flux is absent, the corresponding terminal can be blocked with a *plug* that simply gets or sends a null flux.

Basic parts can also be encapsulated into higher order modules to construct composite devices. They can then be put inside a circuit and connected to other simple or complex parts. For instance, an entire transcription unit may be embedded into a *protein generator* device or a *reporter* device, depending on the kind of protein synthesized (Fig. 1). The design and representation of an intricate network can, hence, be drastically simplified just by putting basic parts, wherever possible, inside composite devices and by connecting these composite devices to the pools, to other basic parts and also between each other when necessary. Finally, the MDL-encoded model for a complete circuit can be directly exported into Matlab code for deterministic simulations. Alternatively, the model can be exported into the more general SBML format (Hucka *et al.*, 2003). After a parsing step through a stand-alone Perl script (due to the specific SBML format generated by ProMoT; see Supplementary Material), one can choose the most appropriate software to run deterministic or stochastic simulations.

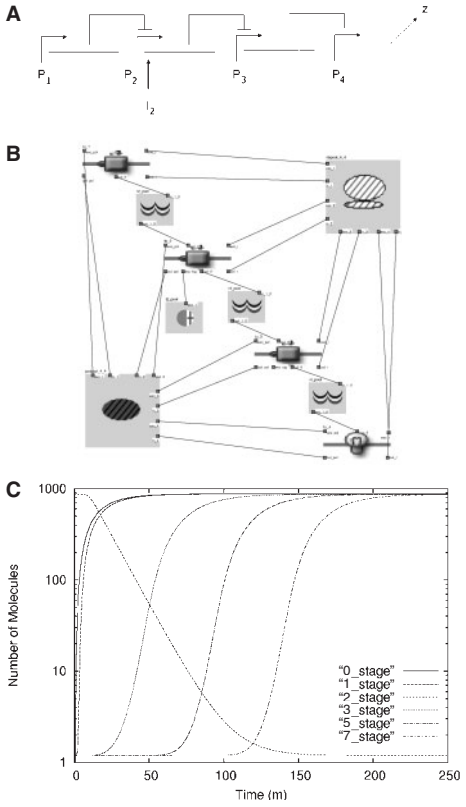


Fig. 2. Engineered cascades. (**A**) Scheme of the three-step cascade. Every stage is lead by a promoter (P_i); the first three genes produce a repressor, the last one a reporter protein (z). I_2 represents the inducer acting on the repressor of promoter P_2. (**B**) Implementation of the three-step cascade with ProMoT. A composite device (protein generator or reporter) is used for each transcription unit. (**C**) Multiple-step-cascade deterministic simulations. Beside the expression levels of some of the cascades between Steps 1 and 7, the single gene expression (0 stage) is shown.

5 RESULTS

For a proof-of-concept study, we tested our modeling framework on some of the best-established synthetic genetic circuits. The results presented in this section have been obtained by running deterministic simulations in COPASI (Hoops *et al.*, 2006) and stochastic simulations in Dizzy (Ramsey *et al.*, 2005), illustrating the compatibility of the concept with different software tools.

As a first benchmark, we chose the seven-step-cascade device (Hooshangi *et al.*, 2005). The simpler three-step cascade is shown in Figure 2A, B. In this circuit, every gene synthesizes a

repressor that acts only on the successive *cis*-regulatory part. In our implementation, we made use of a basic promoter in the first transcription unit. All the others units are controlled by inducible two-operator promoters (see Supplementary Material for details), although only the second-stage promoter is induced by a signal.

To compare simulation results with the stochastic simulations reported in Hooshangi *et al.* (2005), we used the given parameter values and only changed the translation initiation frequency and the leakage transcription rate. Moreover, we extrapolated the association and dissociation constants between RNA polymerases and promoters, and between ribosomes and RBSs from literature data (see Supplementary Material for all details). Following (Hooshangi *et al.*, 2005), every cascade step was reproduced in 20 copies. Simulations were run in two steps: first we let the system reach a steady state in the absence of external signals, then inducers were sent to the second-stage promoter with a fixed rate. Cooperativity between repressors has not been taken into account. Although our deterministic calculations give reporter molecule numbers (from the last gene expression unit) that are slightly lower for the basal production alone, the qualitative behavior of the system is correctly reproduced (Fig. 2C). In particular, the time delay between Steps 3 and 5 (as well as between Steps 5 and 7) is roughly 46 min, which matches well with the 44 min inferred by Hooshangi *et al.* (2005).

As another benchmark we considered the so-called *Repressilator*, a ring oscillator established in bacteria (Elowitz and Leibler, 2000). Following the original publication, we simulated it as a circuit made of three identical transcription units where the first gene represses the second gene, the second represses the third, and this in turn inactivates the first gene (Fig. 3A, B). Stochastic simulations (Fig. 3C) show that for the chosen parameter values, oscillations in the expression of the three repressor genes are sustained for a long time period. A detailed description of the circuit simulation together with a discussion of the RNA polymerase and ribosome dynamics inside this network is provided in Supplementary Material.

Besides these two benchmarks we also realized the *positive and negative feedback oscillator* (Atkinson *et al.*, 2003), the *pulse generating network* (Basu *et al.*, 2004) and the *bistable toggle switch* (Gardner *et al.*, 2000). In all cases, we were able to reproduce their behavior correctly (see Supplementary Material). In addition, we developed an 'artificial' large-scale circuit, which illustrates that even with moderate network complexity, the dependency of the behavior on global pools of, for instance, RNA polymerases is significant; correspondingly, one expects an impact of such circuits on the 'natural' cellular behavior, which needs to be accounted for (see Supplementary Material for details).

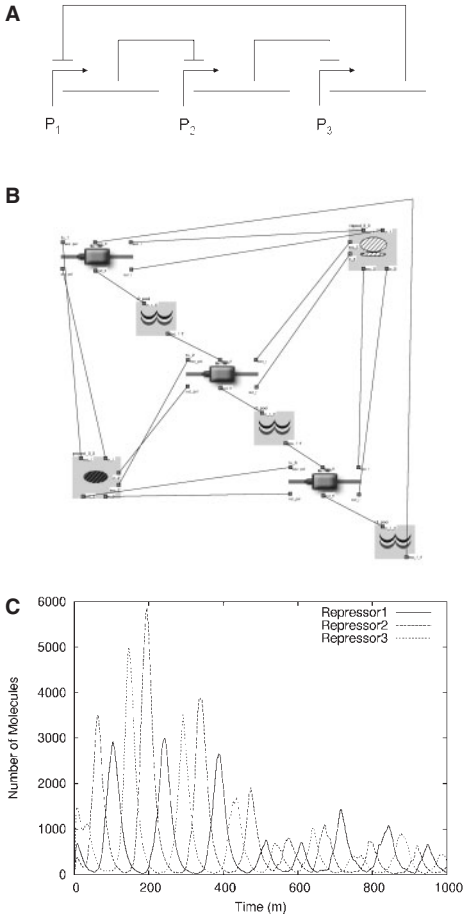


Fig. 3. The repressilator. (**A**) Circuit scheme. (**B**) Implementation with ProMoT. Three protein generators and five pools have been deployed on the canvas. (**C**) Result of stochastic simulations.

6 CONCLUSION

Conceptually, the design of synthetic gene circuits with composable parts has been proposed, but not yet fully realized in a corresponding model-based design tool. Here, we present a formal modeling framework based on the ODE formalism that permits modular model composition. A synthetic circuit can be simulated just by connecting the desired parts to each other. The interfaces between the parts are established by at least four different common signal carriers whose fluxes are exchanged between the parts themselves and the pools where these molecules are stored. To test the validity of the concept, we reproduced the behavior of several well-established synthetic circuits; the simulation results were in good agreement with literature data.

Compared to other methods and tools for synthetic circuit design, our solution appears extremely easy and intuitive to use. It permits building circuits visually, just by displaying the desired parts on the screen and by connecting them through wires. This amounts to basically reproducing circuit schemes without caring about the underlying MDL part code. Starting from the basic parts, one can assemble composite devices of different degree of complexity so that even the design of a network made of dozens of genes is a relatively easy task. Simulations of complex networks, furthermore, can be run without particular constraints because of a detailed description of the reactions taking place inside each part. Compared to the traditional Hill formalism, this enables full scalability. As a consequence, one can directly estimate the value of parameters generally 'hidden' inside the Hill constants and coefficients, and understand their order of magnitude required to yield particular dynamic phenomena such as oscillations. Once the circuit model has been designed, its MDL code serves as a template that can be reloaded and modified in the GUI of ProMoT. Exported into SBML or Matlab format, the circuit model generality is retained. The associated files can be reused for all the necessary simulation studies.

To improve the method, we intend to generalize the promoter construction to enable combinatorial promoter modeling and to include cooperativity phenomena in more detail. More generally, combining the design tool with, for instance, the MIT Registry, literature databases, and other resources could eventually establish a new computational infrastructure for synthetic biology that enables researchers to select biological parts accurately and then to design and test the functioning of the genetic circuits under study in an intuitive, automated fashion.

ACKNOWLEDGEMENTS

We thank D. Müller, M. Terzer, M. Uhr, S. Armstrong, S. Dimopoulos and D. Endy for helpful suggestions and discussions.

Funding: We gratefully acknowledge financial support by the EU project *EMERGENCE* - FP6-NEST contract No. 043338 (http://www.emergence.ethz.ch/).

Conflict of Interest: none declared.

REFERENCES

Alon,U. (2006) *An Introduction to Systems Biology*. Chapman & Hall/CRC Press, Boca Raton, FL.

Andrianantoandro,E. *et al*. (2006) Synthetic biology: new engineering rules for an emerging discipline. *Mol. Syst. Biol.*, **2**, 2006.0028.

Atkinson,M.R. *et al*. (2003) Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in Escherichia coli. *Cell*, **113**, 597–607.

Basu,S. *et al*. (2004) Spatiotemporal control of gene expression with pulse-generating networks. *Proc. Natl Acad. Sci. USA*, **101**, 6355–6360.

Benner,S.A. and Sismour,A.M. (2005) Synthetic biology. *Nat. Rev. Genet.*, **6**, 533–543.

Cai,Y. *et al*. (2007) A syntactic model to design and verify synthetic genetic constructs derived from standard biological parts. *Bioinformatics*, **23**, 2760–2767.

Drubin,D.A. *et al*. (2007) Designing biological systems. *Genes Dev.*, **21**, 242–254.

Elowitz,M.B. and Leibler,S. (2000) A synthetic oscillatory network of transcriptional regulators. *Nature*, **403**, 335–338.

Endy,D. and Deese,I. with the MIT Synthetic Biology Working Group (2005) Adventures in synthetic biology. Appeared in Foundations for engineering biology p449, *Nature*, **438**, 449–453.

Funahashi,A. *et al*. (2003) CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. *BIOSILICO*, **1**, 159–162.

Gardner,T.S. *et al*. (2000) Construction of a genetic toggle switch in Escherichia coli. *Nature*, **403**, 339–342.

Ginkel,M. *et al*. (2003) Modular modeling of cellular systems with ProMoT/Diva. *Bioinformatics*, **19**, 1169–1176.

Goler,J.A. (2004) BioJADE: A design and simulation tool for synthetic biological systems. *Technical report*, MIT, Cambridge, MA.

Hasty,J. *et al*. (2002) Engineered gene circuits. *Nature*, **420**, 224–230.

Hoops,S. *et al*. (2006) COPASI–a COmplex PAthway SImulator. *Bioinformatics*, **22**, 3067–3074.

Hooshangi,S. *et al*. (2005) Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc. Natl Acad. Sci. USA*, **102**, 3581–3586.

Hucka,M. *et al*. (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, **19**, 524–531.

Kærn,M. and Weiss,R. (2006) Synthetic gene regulatory systems. In Szallasi,Z. *et al*. (eds) *System Modeling in Cellular Biology*. The MIT Press, Cambridge, MA, pp. 269–295.

Klipp,E. *et al*. (2005) *Systems Biology in Practice*. WILEY-VCH Verlag, Weinheim.

Kosuri,S. *et al*. (2007) TABASCO: a single molecule, base-pair resolved gene expression simulator. *BMC Bioinformatics*, **8**, 480.

Nagel,L.W. and Pederson,D.O. (1973) SPICE (Simulation Program with Integrated Circuit Emphasis). *Memorandum No. ERL-M382*, University of California, Berkley.

Ramsey,S. *et al*. (2005) Dizzy: stochastic simulation of large-scale genetic regulatory networks. *J. Bioinform. Comput. Biol.*, **3**, 415–436.

Rouilly,V. *et al*. (2007) Registry of BioBrick models using CellML. *BMC Syst. Biol.*, **1**(Suppl. 1), 79–80.

Samoilov,M.S. and Arkin,A.P. (2006) Deviant effects in molecular reaction pathways. *Nat. Biotechnol.*, **24**, 1235–1240.

Sayut,D.J. *et al*. (2007) Engineering and applications of genetic circuits. *Mol. Biosyst.*, **3**, 835–840.

Tomshine,J. and Kaznessis,Y.N. (2006) Optimization of a stochastically simulated gene network model via simulated annealing. *Biophys. J.*, **91**, 3196–3205.

Computational design of synthetic circuits with composable parts.
Supplementary Data

M.A. Marchisio and J. Stelling

Contents

Chapter 1

The repressilator simulation

As explained in the main text, the repressilator is a ring oscillator where activation of the first promoter (P_1) leads to the production of R_2, the repressor of the second promoter (P_2). P_2 represents the cis-regulatory part of the gene that encodes for R_3, that is, the repressor of the third promoter (P_3). To close the ring, the transcription unit containing P_3 produces R_1, which inactivates P_1 (Fig. 1.1A).
This network is composed of three transcription units (12 Registry parts in total) and of pools for polymerases, ribosomes and three transcription factors. Signal plugs block the signal terminal present in every promoter since no inducers enter this part (Fig. 1.2A).
Transcriptional read-through has also been neglected so that the $PoPS^{rt}$ flux and the terminator variable η are absent.
Following [Elowitz and Leibler, 2000] we treat the three transcription units and the three repressors as identical. However, the use of three different pools forces each repressor to act only on one target promoter.
In the following, we describe in detail the reactions which take place in each of the four basic parts (two-operator promoter, RBS, transcription factor coding and terminator) and of the three different pools along with the associated ODEs. We also discuss the dynamics of RNA polymerases and ribosomes inside the network and, finally, we illustrate how we performed stochastic simulations.

1.1 The parts

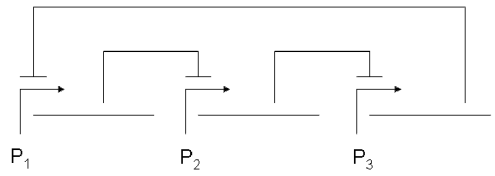
1.1.1 The two-operator promoter

This inducible promoter contains two operators which can be occupied only by repressors of the same type (it is further referred to as promoter_$R_I R_I$). O_1 represents the operator closest to the transcriptional start site. We assume that repressors have the same affinity for O_1 and O_2, independent of the operator position along the promoter sequence. Here, we provide only a partial cooperativity behavior between repressors on the operators: a repressor bound to O_1 does not increase the probability that another repressor will bind to O_2, but it reduces the dissociation frequency of repressor bound to O_2. Degradation of free active repressors (R_j^a, $j = 1, \ldots, 3$) in the corresponding transcription factor pool is taken to occur with the same constant decay rate as for repressors bound to an operator. RNA polymerase is allowed to bind to the promoter only when both operators are free. To simplify the model, we decided to incorporate the polymerase binding to a partially occupied promoter (O_1 free, O_2 occupied) into the leakage signal. The promoter state is described by the variable $O_1^s O_2^s$ where s represents the operator state: free (f) or taken (t).

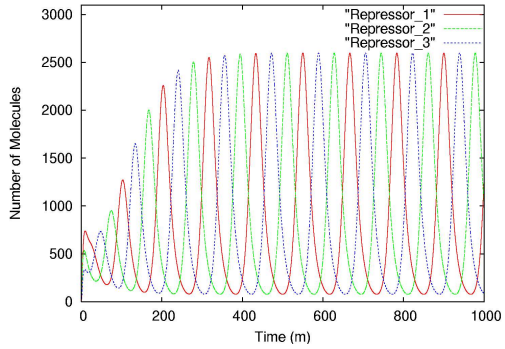
We consider the following possible promoter configurations:

1. $O_1^f O_2^f$, both operators are free;
2. $O_1^t O_2^f$, operator one is occupied preventing RNA polymerase binding;
3. $O_1^f O_2^t$, operator two is occupied preventing RNA polymerase binding;
4. $O_1^t O_2^t$, both operators are occupied preventing RNA polymerase binding;
5. $[Pol O_1^f O_2^f]$, the promoter is occupied by RNA polymerase; mRNA transcription can start.

A



B



C

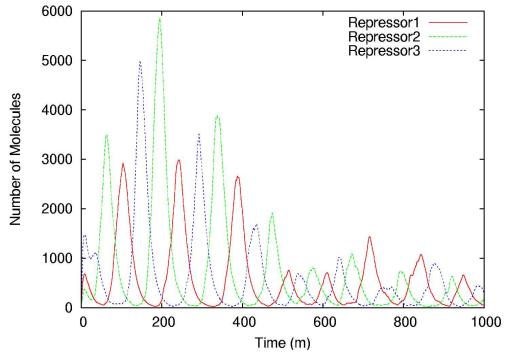


Figure 1.1: The repressilator. (A) Circuit scheme. (B) Deterministic simulation results. (C) Stochastic simulation results.

Symbols

α_1	association rate constant between repressors and O_1
β_1	dissociation rate constant between repressors and O_1, no matter the state of O_2
α_2	association rate constant between repressors and O_2
β_{2f}	dissociation rate constant between repressors and O_2 when O_1 is free
β_{2t}	dissociation rate constant between repressors and O_2 when O_1 is occupied
k_2^{lk}	transcription initiation frequency due to leakage
R_j^a	free active repressor concentrations ($j = 1, \ldots, 3$)
t_{cl}	clearance time

Reactions

$$O_1^t O_2^f \xrightarrow{k_D} O_1^f O_2^f \tag{1.1}$$

$$O_1^f O_2^t \xrightarrow{k_D} O_1^f O_2^f \tag{1.2}$$

$$O_1^t O_2^t \xrightarrow{k_D} O_1^f O_2^t \tag{1.3}$$

$$O_1^t O_2^t \xrightarrow{k_D} O_1^t O_2^f \tag{1.4}$$

$$R_j^a + O_1^f O_2^f \overset{(\alpha_1,\beta_1)}{\rightleftharpoons} O_1^t O_2^f \tag{1.5}$$

$$R_j^a + O_1^f O_2^t \overset{(\alpha_1,\beta_1)}{\rightleftharpoons} O_1^t O_2^t \tag{1.6}$$

$$R_j^a + O_1^t O_2^f \overset{(\alpha_2,\beta_{2t})}{\rightleftharpoons} O_1^t O_2^t \tag{1.7}$$

$$R_j^a + O_1^f O_2^f \overset{(\alpha_2,\beta_{2f})}{\rightleftharpoons} O_1^f O_2^t \tag{1.8}$$

$$Pol^{free} + O_1^f O_2^f \overset{(k_1,k_{-1})}{\rightleftharpoons} [PolO_1^f O_2^f] \xrightarrow{k_2} O_1^f O_2^f + Pol^{cl} \tag{1.9}$$

Equations

$$k_2 = {t_{cl}}^{-1} \tag{1.10}$$

$$O_1^f O_2^f = \{O_1 O_2\}_T - O_1^t O_2^f - O_1^f O_2^t - O_1^t O_2^t - [PolO_1^f O_2^f] \tag{1.11}$$

$$PoPS^b = -k_1 Pol^{free} O_1^f O_2^f + k_{-1}[PolO_1^f O_2^f] - PoPS^{lk} \tag{1.12}$$

$$\begin{aligned} FaPS^b = \quad & - \quad \alpha_1 R_j^a (O_1^f O_2^f + O_1^f O_2^t) + \beta_1 (O_1^t O_2^f + O_1^t O_2^t) + \\ & - \quad \alpha_2 R_j^a O_1^f O_2^f + \beta_{2f} O_1^f O_2^t - \alpha_2 R_j^a O_1^t O_2^f + \beta_{2t} O_1^t O_2^t \end{aligned} \tag{1.13}$$

$$PoPS^{lk} = k_2^{lk} (O_1^f O_2^t + O_1^t O_2^f + O_1^t O_2^t) \tag{1.14}$$

$$PoPS^{out} = k_2 [PolO_1^f O_2^f] + PoPS^{lk} \tag{1.15}$$

$$\frac{d[PolO_1^f O_2^f]}{dt} = k_1 Pol^{free} O_1^f O_2^f - (k_{-1} + k_2)[PolO_1^f O_2^f] \tag{1.16}$$

$$\begin{aligned} \frac{dO_1^f O_2^t}{dt} = \quad & - \quad \alpha_1 R_j^a O_1^f O_2^t + \beta_1 O_1^t O_2^t + \alpha_2 R_j^a O_1^f O_2^f - \beta_{2f} O_1^f O_2^t + \\ & - \quad k_D O_1^f O_2^t + k_D O_1^t O_2^t \end{aligned} \tag{1.17}$$

$$\frac{dO_1^t O_2^f}{dt} = \alpha_1 R_j^a O_1^f O_2^f - \beta_1 O_1^t O_2^f - \alpha_2 R_j^a O_1^t O_2^f + \beta_{2t} O_1^t O_2^t + - k_D O_1^t O_2^f + k_D O_1^t O_2^t \tag{1.18}$$

$$\frac{dO_1^t O_2^t}{dt} = \alpha_1 R_j^a O_1^f O_2^t - \beta_1 O_1^t O_2^t + \alpha_2 R_j^a O_1^t O_2^f - \beta_{2t} O_1^t O_2^t + - 2k_D O_1^t O_2^t \tag{1.19}$$

Parameter values

Parameter	Value	Reference
$\{O_1O_2\}_T$	$2.1 \cdot 10^{-8}\, M$	corresponds to 20 plasmids
k_1	$10^5\, M^{-1}s^{-1}$	[Knaus and Bujard, 1988] and [Lanzer and Bujard, 1988] (tuned)
k_{-1}	$0.01\, s^{-1}$	[Knaus and Bujard, 1988] and [Lanzer and Bujard, 1988] (tuned)
k_2	$0.5\, s^{-1}$	[Elowitz and Leibler, 2000]
k_2^{lk}	$5 \cdot 10^{-5}\, s^{-1}$	[Elowitz and Leibler, 2000] (tuned)
α_1	$10^9\, M^{-1}s^{-1}$	[Elowitz and Leibler, 2000]
α_2	$10^9\, M^{-1}s^{-1}$	[Elowitz and Leibler, 2000]
β_1	$9\, s^{-1}$	[Elowitz and Leibler, 2000]
β_{2f}	$224\, s^{-1}$	[Elowitz and Leibler, 2000]
β_{2t}	$9\, s^{-1}$	[Elowitz and Leibler, 2000]

1.1.2 The RBS

The only difference between the present RBS and the one described in the main text is the absence of a $PoPS^{rt}$ coming from the terminator that belongs to the same transcription unit.

Reactions

$$PoPS^{in} \Longrightarrow [PolB] \xrightarrow{k_{el}^{RBS}} Pol^{el} + B \tag{1.20}$$

$$r^{free} + b \overset{(k_{1r}, k_{-1r})}{\rightleftharpoons} [rb] \xrightarrow{k_{2r}} r^{cl} + b \tag{1.21}$$

$$b \xrightarrow{k_d} \tag{1.22}$$

$$[rb] \xrightarrow{k_d} r^{free} \tag{1.23}$$

Equations

$$k_{el}^{RBS} = \frac{v_{el}}{l_{RBS}} \tag{1.24}$$

$$\frac{d[PolB]}{dt} = PoPS^{in} - k_{el}^{RBS}[PolB] \tag{1.25}$$

$$PoPS^{out} = k_{el}^{RBS}[PolB] \tag{1.26}$$

$$k_{2r} = (t_{cl}^r)^{-1} \tag{1.27}$$

$$\frac{db}{dt} = k_{el}^{RBS}[PolB] - k_{1r} r^{free} b + (k_{-1r} + k_{2r})[rb] - k_d b \tag{1.28}$$

$$\frac{d[rb]}{dt} = k_{1r} r^{free} b - (k_{-1r} + k_{2r} + k_d)[rb] \tag{1.29}$$

$$RiPS^b = -k_{1r} r^{free} b + (k_{-1r} + k_d)[rb] \tag{1.30}$$

$$RiPS^{out} = k_{2r}[rb] \tag{1.31}$$

Parameter values

Parameter	Value	Reference
k_{1r}	$10^6\, M^{-1}s^{-1}$	[Calogero et al., 1988] (tuned)
k_{-1r}	$0.01\, s^{-1}$	[Calogero et al., 1988] (tuned)
k_{2r}	$0.02\, s^{-1}$	[Tomsic et al., 2000]
k_d	$0.0058\, s^{-1}$	[Elowitz and Leibler, 2000]
v_{el}	$40\, nt/s$	[Kennell and Riezman, 1977]
k_{el}^{RBS}	$2\, s^{-1}$	assuming an average length of 20 nt

1.1.3 The transcription factor coding

This part corresponds to the one described in the paper (see "The protein coding part"). However, here the total amount of synthesized protein (z) is not separately accounted for because the repressors produced are sent to their pool directly.

Reactions

$$PoPS^{in} \Longrightarrow [PolA] \xrightarrow{k_{el}^{PC}} Pol^{el} + A \tag{1.32}$$

$$RiPS^{in} \Longrightarrow [ra] \xRightarrow{k_{el}^{r}} RiPS^{PC} \tag{1.33}$$

$$RiPS^{PC} \Longrightarrow [ru] \xrightarrow{\zeta_r} r^{free} + u \tag{1.34}$$

Equations

$$k_{el}^{PC} = \frac{v_{el}}{l_{PC}} \tag{1.35}$$

$$\frac{d[PolA]}{dt} = PoPS^{in} - k_{el}^{PC}[PolA] \tag{1.36}$$

$$PoPS^{out} = k_{el}^{PC}[PolA] \tag{1.37}$$

$$k_{el}^{r} = \frac{v_{el}^{r}}{l_{PC}} \tag{1.38}$$

$$RiPS^{PC} = k_{el}^{r}[ra] \tag{1.39}$$

$$FaPS^{out} = RiPS^{PC} \tag{1.40}$$

$$\frac{d[ra]}{dt} = RiPS^{in} - k_{el}^{r}[ra] \tag{1.41}$$

$$\frac{d[ru]}{dt} = RiPS^{PC} - \zeta_r[ru] \tag{1.42}$$

$$RiPS^{out} = \zeta_r[ru] \tag{1.43}$$

Parameter values

Parameter	Value	Reference
k_D	$0.00116\, s^{-1}$	[Elowitz and Leibler, 2000]
length	620 nt	Registry part BBa_C0040
v_{el}^{r}	$35\, nt/s$	[Kennell and Riezman, 1977]
v_{el}	$40\, nt/s$	[Kennell and Riezman, 1977]
k_{el}^{PC}	$0.064\, s^{-1}$	derived from v_{el} and length
k_{el}^{r}	$0.056\, s^{-1}$	derived from v_{el}^{r} and length
ζ_r	$0.5\, s^{-1}$	[Freistroffer et al., 1997]

1.1.4 The terminator

This terminator differs from the general one presented in the paper because it is not able to produce a $PoPS^{rt}$ signal.

Reactions

$$PoPS^{in} \Longrightarrow [PolT] \tag{1.44}$$

$$[PolT] \xrightarrow{\zeta} Pol^{free} + T \tag{1.45}$$

Equations

$$\frac{d[PolT]}{dt} = PoPS^{in} - \zeta[PolT] \tag{1.46}$$

$$PoPS^{out} = \zeta[PolT] \tag{1.47}$$

Parameter values

Parameter	Value	Reference
ζ	$31.25\, s^{-1}$	[Arkin et al., 1998]

1.2 The pools

1.2.1 The polymerase pool

This pool is connected to each of the three two-operator promoters and terminators present in the network.

Symbols

$PoPS_j^{in}$	incoming $PoPS$ from each terminator ($j = 1, \ldots, 3$)
$PoPS_j^{b}$	incoming $PoPS$ from each promoter ($j = 1, \ldots, 3$)

Reactions

$$PoPS_j^{in} \Longrightarrow Pol^{free} \tag{1.48}$$

$$PoPS_j^{b} \Longrightarrow Pol^{free} \tag{1.49}$$

Equations

$$\frac{dPol^{free}}{dt} = \sum_{j=1}^{3}(PoPS_j^{in} + PoPS_j^{b}) \tag{1.50}$$

Parameter values

Parameter	Value	Reference
$Pol^{free}(t = 0)$	$2.1 \cdot 10^{-6}\, M$	[Lewin, 2000] (tuned)

1.2.2 The ribosome pool

This pool is connected to each of the three RBS and transcription factor coding parts present in the network.

Symbols

$RiPS_j^{in}$	incoming $RiPS$ from each transcription factor coding ($j = 1, \ldots, 3$)
$RiPS_j^{b}$	incoming $RiPS$ from each RBS ($j = 1, \ldots, 3$)

Reactions

$$RiPS_j^{in} \Longrightarrow r^{free} \tag{1.51}$$

$$RiPS_i^{b} \Longrightarrow r^{free} \tag{1.52}$$

Equations

$$\frac{dr^{free}}{dt} = \sum_{j=1}^{3}(RiPS_j^{in} + RiPS_j^{b}) \tag{1.53}$$

Parameter values

Parameter	Value	Reference
$r^{free}(t=0)$	$4.2 \cdot 10^{-6}\, M$	[Lewin, 2000] and [Bremer and Dennis, 1996] (tuned)

1.2.3 The transcription factor pool

Three transcription factor pools with identical parameter values are present in the circuit. Each of them is connected to two basic parts: the transcription factor coding part, which produces the repressors stored in the pool, and the promoter where the repressors interact with the operators. In one of the three pools, the initial concentration of the repressors in its monomeric form has to be different from zero in order to break the stationary steady state of the system and to give rise to oscillations.

Reactions

$$FaPS^{in} \Longrightarrow F^{m} \tag{1.54}$$

$$FaPS^{b} \Longrightarrow F^{free} \tag{1.55}$$

$$F^{m} \xrightarrow{k_D} ; F^{free} \xrightarrow{k_D} \tag{1.56}$$

$$2F^{m} \overset{(\delta,\epsilon)}{\rightleftharpoons} F^{free} \tag{1.57}$$

Equations

$$\frac{dF^{m}}{dt} = 2(-\delta F^{m\,2} + \epsilon F^{free}) - k_D F^{m} + FaPS^{in} \tag{1.58}$$

$$\frac{dF^{d}}{dt} = \delta F^{m\,2} - \epsilon F^{free} - k_D F^{free} + FaPS^{b} \tag{1.59}$$

Parameter values

Parameter	Value	Reference
δ	$10^9\, M^{-1} s^{-1}$	[Tuttle et al., 2005]
ϵ	$10\, s^{-1}$	[Tuttle et al., 2005]
k_D	$0.00116\, s^{-1}$	[Elowitz and Leibler, 2000]
$F^{m}(t=0)$	$5 \cdot 10^{-8}\, M$	taken small in comparison with free RNA polymerase initial concentration

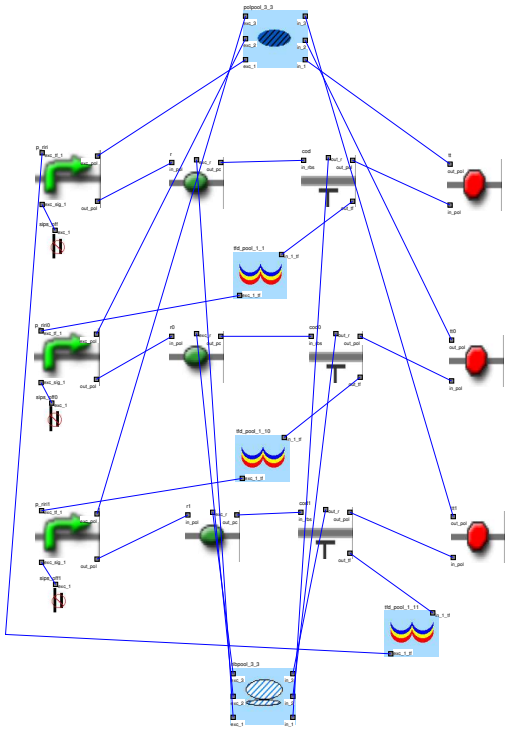
1.2.4 The signal plugs

Every two-operator promoter used in this circuit can receive a flux of inducers which inactivate the repressors. Since this network needs no inducers to work, every promoter terminal normally connected to a signal pool is connected to a signal plug instead. The signal plug represents a simplified pool where the total inducer concentration is equal to zero.

Equations

$$S^{free} = 0 \tag{1.60}$$

A



B

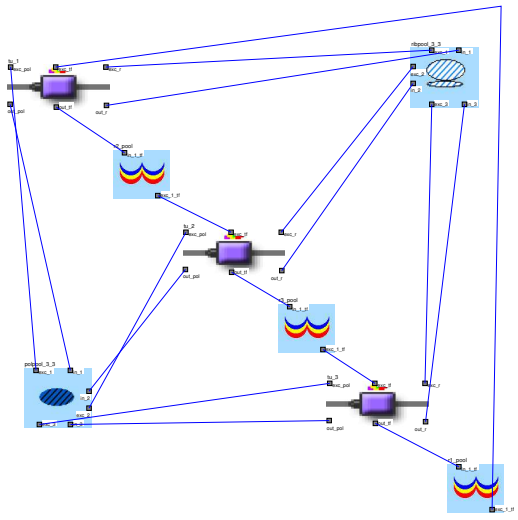


Figure 1.2: The repressilator implementation. (A) Detailed model structure where all Registry parts, pools and plugs are shown. (B) Modular composition, where each transcription unit is represented by a composite device (the protein generator).

1.3 RNA polymerase dynamics

To investigate the impact of the engineered circuit on general cellular dynamics and vice versa, we investigated the RNA polymerase and ribosome dynamics in detail for the repressilator. This analysis, moreover, allowed for a consistency check of the composite model structure because the total pool sizes need to be constant by construction of the model.

Results for RNA polymerase dynamics at different locations in the cell are summarized in Fig. 1.3. From Fig. 1.3A it is evident that once a condition of stable oscillations has been reached, approximately 23 molecules of RNA polymerase (out of a total of about 2023) are required for the transcription of one gene, while access to the other two genes is blocked. Most of these molecules are found inside the transcription factor coding region (see Fig. 1.3D). This is due, in part, to the low affinity between RNA polymerases and the promoter and mainly to the high value of the transcription initiation frequency k_2.

By multiplying the concentration value of every complex made of RNA polymerases and a DNA part ($[PolO_1^f O_2^f]$, $[PolB]$,$[PolA]$ and $[PolT]$) with the corresponding "moving rate" (k_2, k_{el}^{RBS}, k_{el}^{PC} and ζ) we obtain that, when a gene is transcribed, a flux of about 1.2 polymerases flows along a transcription unit. The relative error in model predictions for the total RNA polymerase concentration is shown in Fig. 1.4. These relative errors are on the order of magnitude of the numerical accuracy used in simulation. This indicates that the detailed mass balances across the model are accurate.

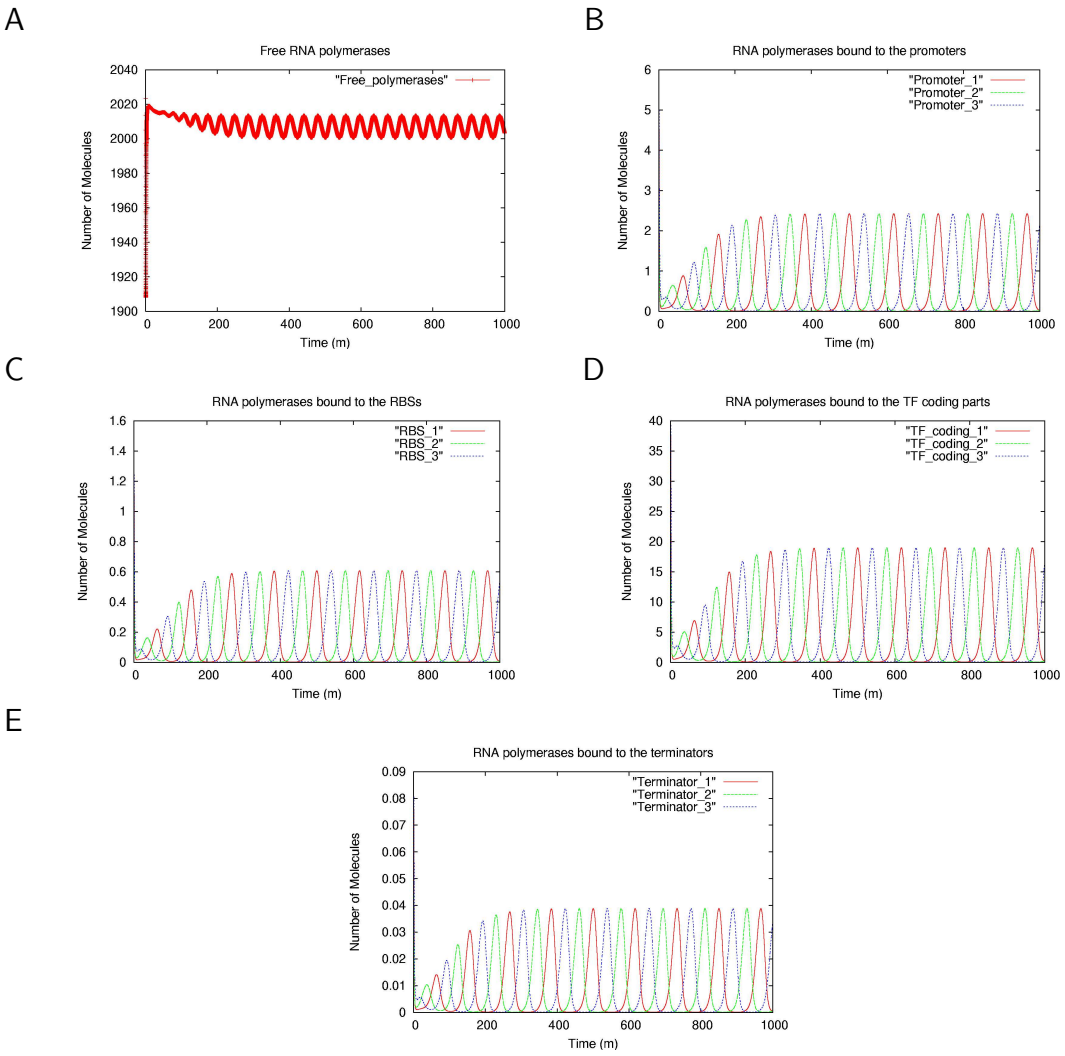


Figure 1.3: RNA polymerase dynamics.

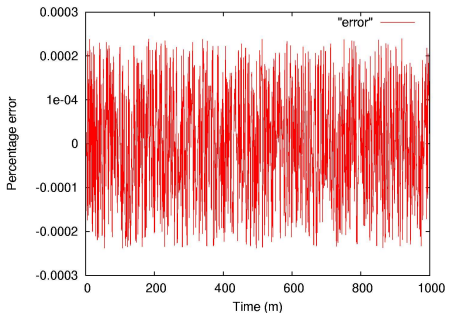


Figure 1.4: Percentage error of the total amount of RNA polymerases.

1.4 Ribosome dynamics

Looking at the ribosome dynamics we can see that at steady state roughly 293 molecules are involved in the translation process (Fig. 1.5A), whereas the total amount of free ribosomes is approximately 4046 molecules. In this case, the majority of the ribosomes (about 202) reside inside the RBS: the value of k_{2r} is lower than k_{el}^{PC} so that, due to the ribosome flux conservation, the total concentration of $[rb]$ must exceed that of $[ra]$. A little more than 72 ribosomes are bound to the translation start site and just about 8 at the stop codon. The rest of the translating molecules are bound to the mRNA of the other two repressors expressed at low levels. The flux of ribosomes along the mRNA can be estimated (as we did above for RNA polymerase) in about 4 molecules per second. Again, we find mass conservation for the total ribosome concentration within the range of numerical accuracy (Fig. 1.6).

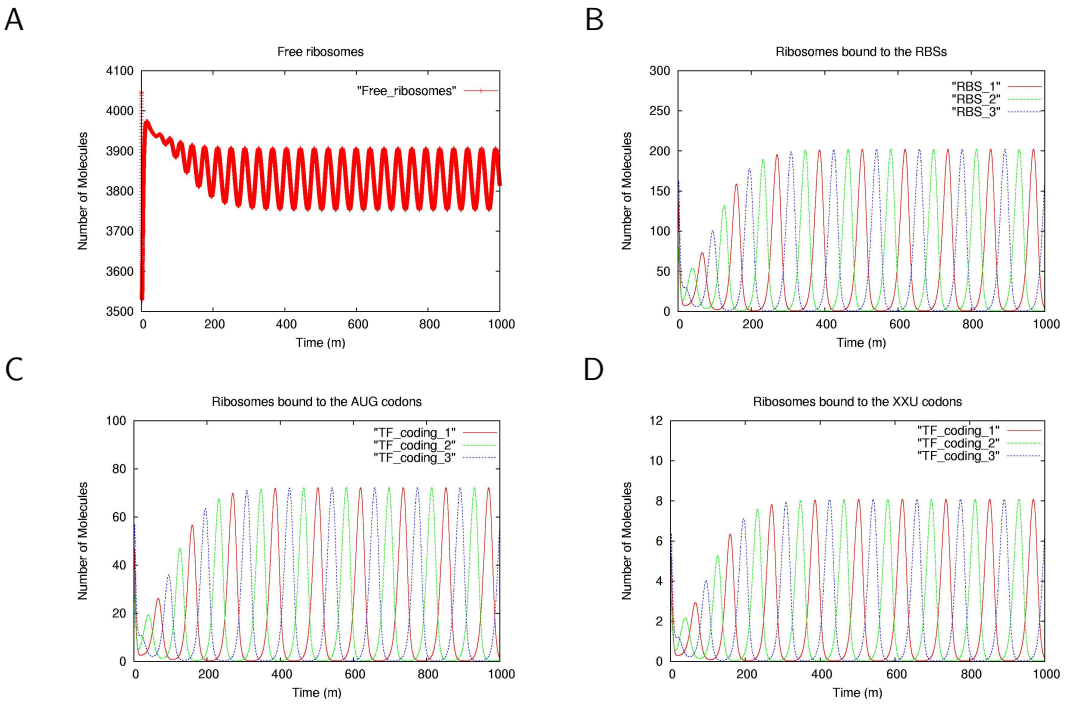


Figure 1.5: Ribosome dynamics.

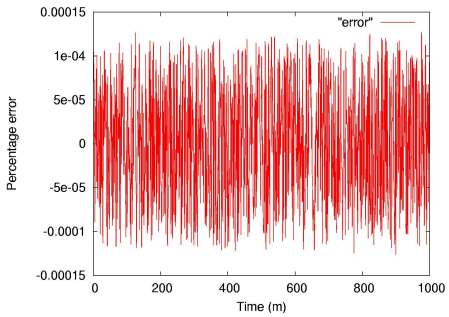


Figure 1.6: Percentage error of the total amount of ribosomes.

1.5 Stochastic simulations

As explained in the paper (see "Implementation"), the MDL code of each part contains *variables* and *equations* (other than terminals). The former can be biochemical compounds and kinetic parameters, the latter are the ODEs associated with every state. ProMoT exports MDL into an SBML format where every variable becomes a *global parameter* and every equation a *rate rule*. All right-hand sides of the ODEs are expressed in equation (MathML). Such an SBML can be read by specific tools such as COPASI. It allows to run deterministic simulations, but the specific SBML format lacks the definition of *species* and *reactions* that are necessary to execute stochastic simulations.
In order to have a general SBML code associated with a circuit designed with ProMoT, we wrote a parser (in Perl) that reads the ProMoT-generated SBML and converts it into a "standard" format. Reaction rate laws compatible with stochastic simulations (e.g., mass-action kinetics) are implicitly assumed by the parser; more specific rate laws (e.g., Michaelis-Menten type laws) will not be recognized.
The parser first reconstructs the concentration and flux (reaction) vectors from the list of global parameters. Then, by processing all the rates rules (ODEs), it builds the stoichiometric matrix. From the columns of the stoichiometric matrix, each reaction can be deduced in terms of its *reactants* and *products* and the corresponding "reaction" tag can be written to a new SBML file. Depending on the application one wants to use for circuit simulation, the parser can generate both a Level-1 and a Level-2 SBML.
For the specific simulations of the repressilator (see main text), note that parameter values of the deterministic model had to be adjusted for the stochastic setting. The corresponding parameter values are provided below.

Modified parameter values

Parameter	**Value**	**Note**
$\{O_1O_2\}_T$	$20\,mlc$	mlc stands for molecule
$Pol^{free}(t=0)$	$2023\,mlc$	converted value
$r^{free}(t=0)$	$4046\,mlc$	converted value
$F^m(t=0)$	$48\,mlc$	converted value (only for one repressor)
k_1	$10^{-4}\,mlc^{-1}s^{-1}$	converted value
k_{1r}	$10^{-3}\,mlc^{-1}s^{-1}$	converted value
k_{2r}	$0.08\,s^{-1}$	set equal to the half of the value in [Elowitz and Leibler, 2000]
α_1	$1\,mlc^{-1}s^{-1}$	converted value
α_2	$1\,mlc^{-1}s^{-1}$	converted value
δ	0	no dimerization has been taken into account
ϵ	0	no dimerization has been taken into account

Chapter 2

Large network dynamics

An important consideration of any engineering design concept is the scalability of the approach for large circuits. Our approach allows to estimate the *scalability* of synthetic gene circuits with respect to the number of available signal carriers because we explicitly model (and, hence, can fix) the quantity of free RNA polymerases and ribosomes.
As an example, we consider an artificial repressilator made of three composite devices (Fig. 2.2). Each composite device realizes a seven-step cascade (Fig. 2.3). This network consists of 24 transcription units. Since every transcription unit is present in 20 copies in the cell, the circuit encompasses a total of 480 gene copies. While the size of the circuit, hence, is not unrealistic, it comprises 10% of the gene copy number of an *E. coli* cell.
Here, we test three scenarios: Both the initial quantity of RNA polymerases and ribosomes are supposed to assume three possible values indicated as *free*, *total* and *infinite*. The *free* concentrations are those used in all our simulations; they correspond to $2.1\,10^{-6}M$ for RNA polymerases and to $4.2\,10^{-6}M$ for ribosomes [Lewin, 2000, Bremer and Dennis, 1996]. Considering also the transcribing/translating molecules, we obtain the *total* concentrations which are estimated to be about $7.35\,10^{-6}M$ for RNA polymerases and about $2.1\,10^{-5}M$ for ribosomes [Lewin, 2000]. *Infinite* concentrations are simply unrealistically huge values, such that the DNA quantity of our network is extremely low in comparison. We have chosen $2.1\,10^{-4}M$ for RNA polymerases and $2.1\,10^{-3}M$ for ribosomes.

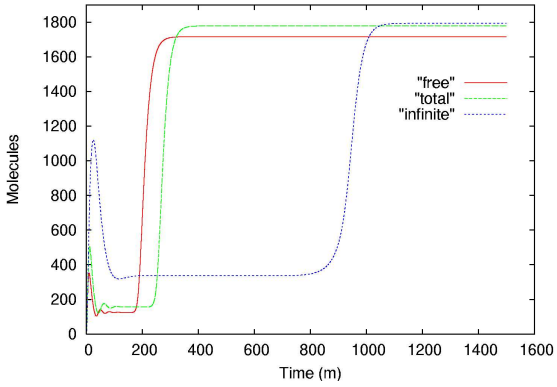


Figure 2.1: R_3 time courses for different settings of RNA polymerase and ribosome pools.

In Fig. 2.1 the amount of the repressors produced by the last gene of the third device (R_3), which acts on the first promoter of the first seven-step cascade, is shown for the three different pairs of concentrations described above. As expected, the network responds quite differently in each of the three cases. Correspondingly, for limited realistic amounts of RNA polymerase and ribosomes, the synthetic circuit has a significant impact on the available pool sizes. Side-effects on natural cellular functions are to be expected from this. Hence, traditional methods that neglect the RNA polymerase and ribosome concentration are not suitable for the scalable study of complex networks, which strongly limits their applicability.

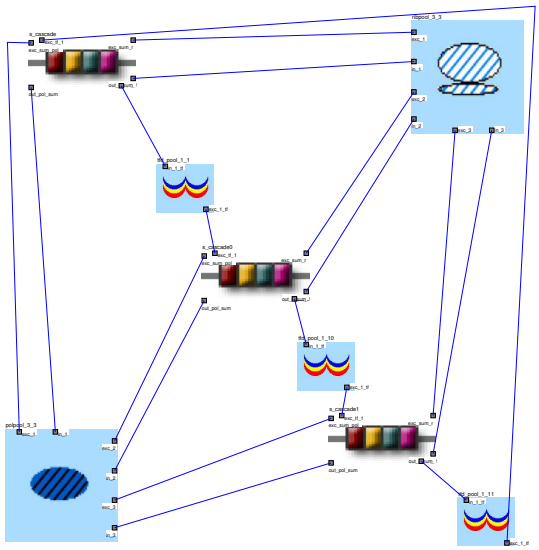


Figure 2.2: Repressilator containing three composite devices.

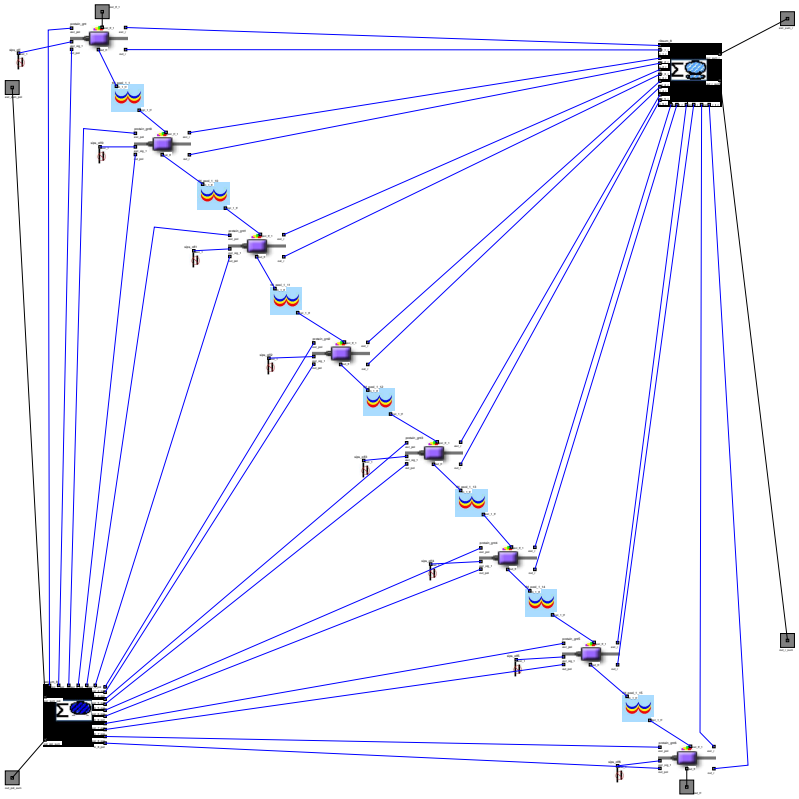


Figure 2.3: Content of each composite device. The black parts on the top-right and the bottom-left communicate directly with the ribosome and the polymerase pool, respectively. They send the total-device value of $PoPS^b$ and $RiPS^b$ and get the amount of Pol^{free} and r^{free}.

Chapter 3

Circuits

3.1 Common parameter values

If not specified otherwise, the following parameter values associated with different basic parts are assumed for every circuit presented here.

(basic) promoter

Parameter	Value	Reference
P_T	$2.1 \cdot 10^{-8}\, M$	[Hooshangi et al., 2005] (correspond to 20 plasmids)
k_1	$10^6\, M^{-1} s^{-1}$	[Lanzer and Bujard, 1988] and [Tuttle et al., 2005] (tuned)
k_{-1}	$0.01\, s^{-1}$	[Lanzer and Bujard, 1988] and [Tuttle et al., 2005] (tuned)
k_2	$0.03\, s^{-1}$	[Hooshangi et al., 2005]

RBS

Parameter	Value	Reference
k_{1r}	$10^6\, M^{-1} s^{-1}$	[Calogero et al., 1988] (tuned)
k_{-1r}	$0.01\, s^{-1}$	[Calogero et al., 1988] (tuned)
k_{2r}	$0.02\, s^{-1}$	[Tomsic et al., 2000] (tuned)
k_d	$0.0116\, s^{-1}$	[Hooshangi et al., 2005]
k_{el}^{RBS}	$2\, s^{-1}$	see the repressilator

transcription factor coding

Parameter	Value	Reference
k_D	$0.00116\, s^{-1}$	[Hooshangi et al., 2005]
k_{el}^{PC}	$0.064\, s^{-1}$	see the repressilator
k_{el}^{r}	$0.056\, s^{-1}$	see the repressilator
ζ_r	$0.5\, s^{-1}$	[Freistroffer et al., 1997]
length	620 nt	Registry part BBa_C0040

reporter coding

Parameter	Value	Reference
k_D	$0.00116\, s^{-1}$	[Hooshangi et al., 2005]
k_{el}^{PC}	$0.056\, s^{-1}$	corresponding to a $v_{el} = 40nt/s$ [Kennell and Riezman, 1977]
k_{el}^{r}	$0.049\, s^{-1}$	corresponding to a $v_{el} = 35nt/s$ [Kennell and Riezman, 1977]
ζ_r	$0.5\, s^{-1}$	[Freistroffer et al., 1997]
length	710 nt	Registry part BBa_I15017

terminator

Parameter	Value	Reference
ζ	$31.25\,s^{-1}$	[Arkin et al., 1998]
η	0	

polymerase pool

Parameter	Value	Reference
$Pol^{free}(t=0)$	$2.1 \cdot 10^{-6}\,M$	[Lewin, 2000] (tuned)

ribosome pool

Parameter	Value	Reference
$r^{free}(t=0)$	$4.2 \cdot 10^{-6}\,M$	[Lewin, 2000] and [Bremer and Dennis, 1996] (tuned)

transcription factor pool

Parameter	Value	Reference
δ	$5 \cdot 10^{5}\,M^{-1}s^{-1}$	[Hooshangi et al., 2005]
ϵ	$1.7 \cdot 10^{-5}\,s^{-1}$	[Hooshangi et al., 2005]
k_D	$0.00116\,s^{-1}$	[Hooshangi et al., 2005]

signal pool

Parameter	Value	Reference
k	$3 \cdot 10^{-9}\,Ms^{-1}$	[Hooshangi et al., 2005] (tuned on the total inducer concentration)
k_D	$3.21 \cdot 10^{-5}\,s^{-1}$	aTC decay rate

3.2 The seven-step cascade

promoter_$R_I R_I$

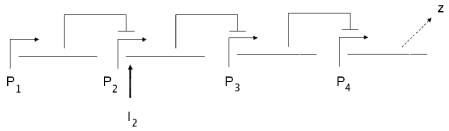
This is an inducible two-operator promoter whose operators host the same kind of repressors. Each repressor is inactivated by n inducers.

Parameter	Value	Reference
$\{O_1O_2\}_T$	$2.1 \cdot 10^{-8}\,M$	[Hooshangi et al., 2005]
k_2	$0.03\,s^{-1}$	[Hooshangi et al., 2005]
k_2^{lk}	$3.3 \cdot 10^{-5}\,s^{-1}$	[Hooshangi et al., 2005] (tuned)
α_1	$3.3 \cdot 10^{6}\,M^{-1}s^{-1}$	[Hooshangi et al., 2005]
α_{2f}	$3.3 \cdot 10^{6}\,M^{-1}s^{-1}$	[Hooshangi et al., 2005]
α_{2t}	$3.3 \cdot 10^{6}\,M^{-1}s^{-1}$	[Hooshangi et al., 2005]
β_1	$0.0017\,s^{-1}$	[Hooshangi et al., 2005]
β_{2f}	$0.0017\,s^{-1}$	[Hooshangi et al., 2005]
β_{2t}	$0.0017\,s^{-1}$	[Hooshangi et al., 2005]
γ	$8.3 \cdot 10^{5}\,M^{-1}s^{-1}$	[Hooshangi et al., 2005] (taken equal to λ)
λ	$8.3 \cdot 10^{5}\,M^{-1}s^{-1}$	[Hooshangi et al., 2005]
μ	$0.0017\,s^{-1}$	[Hooshangi et al., 2005]
n	1	[Hooshangi et al., 2005]

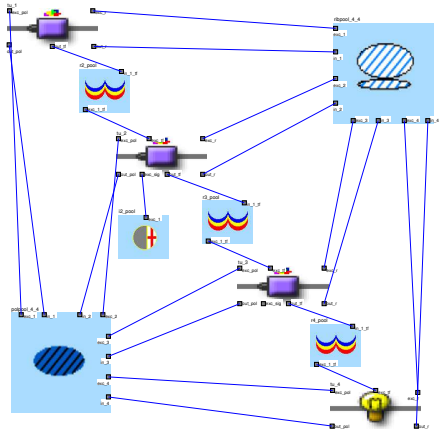
3.3 The mixed positive and negative feedback oscillator

Being an oscillating system, most of its parameter values have been chosen according to the reference values for the repressilator.

A



B



C

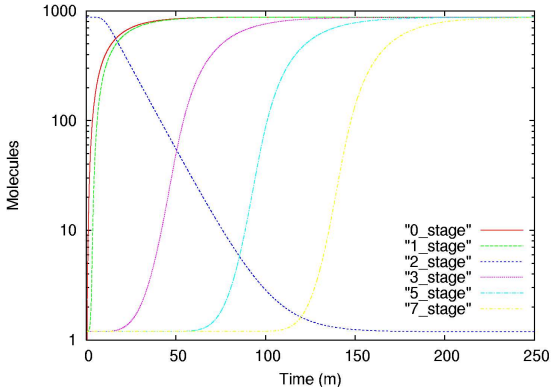


Figure 3.1: The three-step-cascade network. (A) Scheme. (B) Implementation. (C) Results for the seven-step cascade.

promoter_$A_C R_I$

This is a two-operator promoter. The O_1 operator hosts one repressor, which could be inactivated by n_1 inducers, whereas the O_2 operator hosts one activator, which could be inactivated by n_2 corepressors.

Parameter	Value	Reference
$\{O_1O_2\}_T$	$2.1 \cdot 10^{-8}\, M$	corresponds to 20 plasmids
k_2	$0.5\, s^{-1}$	[Elowitz and Leibler, 2000]
k_2^{lk}	$10^{-7}\, s^{-1}$	tuned starting from the value in [Elowitz and Leibler, 2000]
α_1	$10^9\, M^{-1} s^{-1}$	[Elowitz and Leibler, 2000] (association rate constant between an activator and O_1)
α_2	$10^9\, M^{-1} s^{-1}$	[Elowitz and Leibler, 2000] (association rate constant between a repressor and O_2)
β_1	$0.0017\, s^{-1}$	[Hooshangi et al., 2005] (dissociation rate constant from O_1)
β_2	$0.0017\, s^{-1}$	[Hooshangi et al., 2005] (dissociation rate constant from O_2)
γ_1,γ_2	0	
λ_1,λ_2	0	
μ_1,μ_2	0	
n_1,n_2	0	

promoter_A_C

This is a repressible one-operator promoter. Every activator could be inactivated by n corepressors.

Parameter	Value	Reference
P_T	$2.1 \cdot 10^{-8}\, M$	corresponds to 20 plasmids
k_2	$0.5\, s^{-1}$	[Elowitz and Leibler, 2000]
k_2^{lk}	$10^{-7}\, s^{-1}$	tuned starting from the value in [Elowitz and Leibler, 2000]
α	$10^9\, M^{-1} s^{-1}$	[Elowitz and Leibler, 2000]
β	$0.0017\, s^{-1}$	[Hooshangi et al., 2005]
γ	0	
λ	0	
μ	0	
n	0	

RBS_1

Parameter	Value	Reference
k_{2r}	$0.5\, s^{-1}$	set much higher than in RBS_2

RBS_2

Parameter	Value	Reference
k_{2r}	$0.015\, s^{-1}$	tuned

transcription factor coding

Parameter	Value	Reference
$z(t=0)$	$10^{-8}\, M$	initial concentration both of repressors and activators; tuned from the repressilator

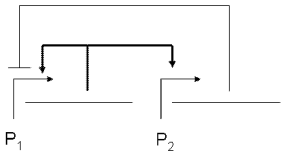
transcription factor pool

Parameter	Value	Reference
$F^m(t=0)$	$10^{-8}\, M$	see above the "transcription factor coding"

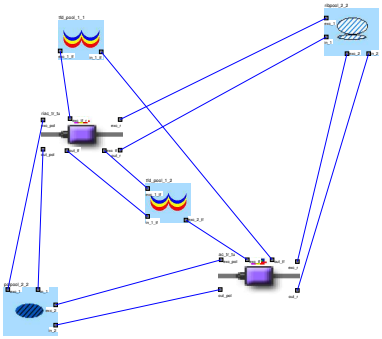
3.4 The pulse generating network

Reminiscent of a step-cascade circuit, most of its parameter values have been taken from the seven-step cascade network.

A



B



C

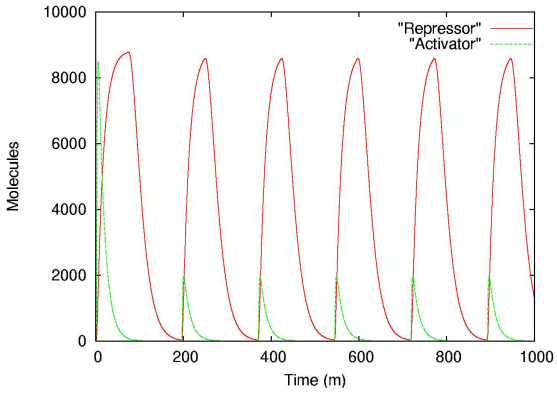


Figure 3.2: The mixed positive and negative feedback oscillator. (A) Scheme: the same kind of activators (bold arrows) binds both to the two-operator promoter P_1, which regulates the production of the activators themselves, and to the one-operator promoter P_2, which regulates the production of repressors capable of inhibiting the activity of P_1. Initially, a small amount of the activators stimulates a high production of the activators followed, with a small time delay, by the repressor expression. This limits the activator production and causes the oscillating behavior. (B) Implementation. (C) Results.

promoter_A_I

This is an inducible one-operator promoter where every activator is activated by n inducers.

Parameters	Values	Reference
P_T	$2.1 \cdot 10^{-8}\, M$	corresponds to 20 plasmids
k_2	$0.01 \div 1.5\, s^{-1}$	values chosen to have significant plots
k_2^{lk}	0	
α	$10^5\, M^{-1} s^{-1}$	tuned starting from the value in [Hooshangi et al., 2005]
β	$0.001\, s^{-1}$	tuned starting from the value in [Hooshangi et al., 2005]
λ	$8.3 \cdot 10^5\, M^{-1} s^{-1}$	[Hooshangi et al., 2005]
μ	$0.0017\, s^{-1}$	[Hooshangi et al., 2005]
n	1	[Hooshangi et al., 2005]

promoter_$A_I R_I$

This is an inducible two-operator promoter whose operators host one activator (activated by n_2 inducers) and one repressor (which could be inactivated by n_1 inducers).

Parameters	Values	Reference
$\{O_1 S_1\}_T$	$2.1 \cdot 10^{-8}\, M$	corresponds to 20 plasmids
k_2	$0.03\, s^{-1}$	[Hooshangi et al., 2005]
k_2^{lk}	0	
α_1	$3.3 \cdot 10^6\, M^{-1} s^{-1}$	[Hooshangi et al., 2005]
α_2	$3.3 \cdot 10^6\, M^{-1} s^{-1}$	[Hooshangi et al., 2005]
β_1	$0.0017\, s^{-1}$	[Hooshangi et al., 2005]
β_2	$0.0017\, s^{-1}$	[Hooshangi et al., 2005]
γ_2	0	see the Seven-step cascade for its meaning
λ_1	0	
λ_2	$8.3 \cdot 10^5\, M^{-1} s^{-1}$	[Hooshangi et al., 2005] (association rate constant between n_2 inducers and one activator)
μ_1	0	
μ_2	$0.0017\, s^{-1}$	[Hooshangi et al., 2005] (dissociation rate constant of n_2 inducers from one activator)
n_1	0	
n_2	1	[Hooshangi et al., 2005]

signal pool

Parameters	Values	Reference
k	$5 \cdot 10^{-12} \div 5 \cdot 10^{-9}\, M s^{-1}$	values chosen to have significant plots

3.5 The bistable toggle switch

We used the parameter values in [Hooshangi et al., 2005] to reproduce the circuit qualitative behavior.

promoter_R_I

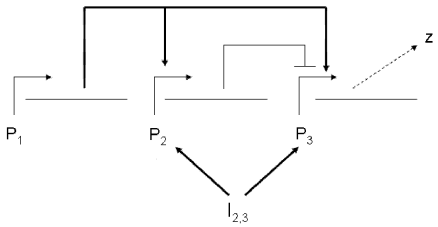
This is an inducible one-operator promoter, like the one illustrated in the paper.

Parameters	Values	Reference
$\{O_1 O_2\}_T$	$2.1 \cdot 10^{-8}\, M$	corresponds to 20 plasmids
k_2	$0.03\, s^{-1}$	[Hooshangi et al., 2005]
k_2^{lk}	$10^{-5}\, s^{-1}$	[Hooshangi et al., 2005] (tuned)
α	$3.3 \cdot 10^6\, M^{-1} s^{-1}$	[Hooshangi et al., 2005]
β	$0.0017\, s^{-1}$	[Hooshangi et al., 2005]
γ, λ	$8.3 \cdot 10^5\, M^{-1} s^{-1}$	[Hooshangi et al., 2005]
μ	$0.0017\, s^{-1}$	[Hooshangi et al., 2005]
n	1	[Hooshangi et al., 2005]

signal pool

Parameters	Values	Reference
k	$1 \cdot 10^{-9}\, M s^{-1}$	[Hooshangi et al., 2005] (tuned)

A



B

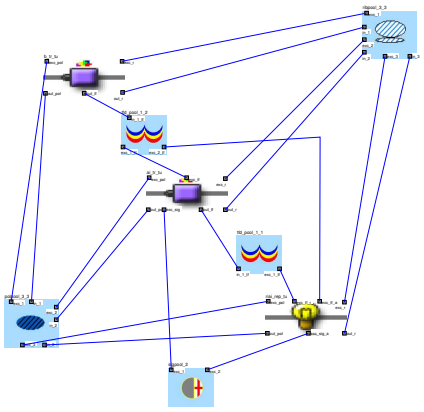
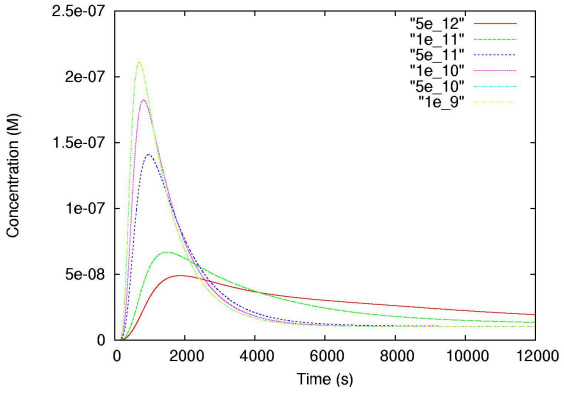


Figure 3.3: The pulse generating network. (A) Scheme: an activator (bold arrows in the upper part of the graph) stimulates the production both of a reporter protein (z) and of a repressor which in turn inhibits the reporter protein expression. The pulse in the production of z is due to a time delay between the activator and the repressor synthesis. Each activator dimer,furthermore, needs to be activated by one inducer molecule ($I_{2,3}$). Note also that P_1 is a basic promoter. (B) Implementation.

A



B

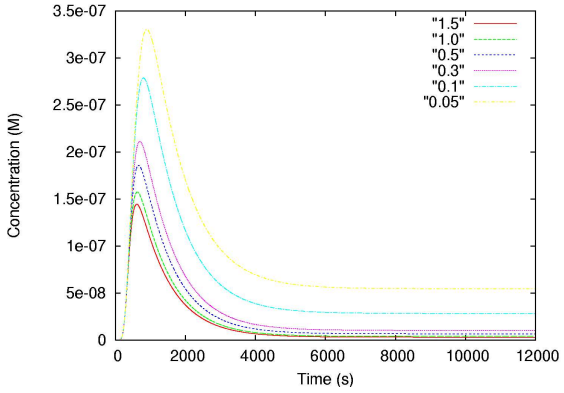
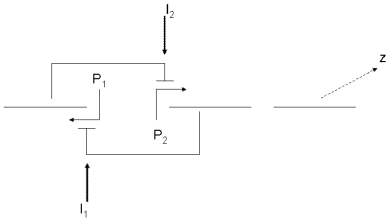
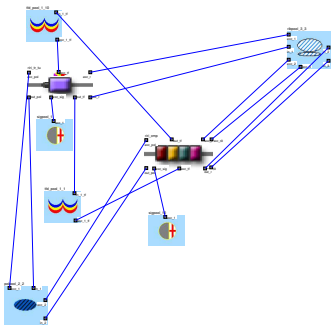


Figure 3.4: The pulse generating network: results (A) Network response for different values of the inducer production rate (k) with$k_2 = 0.3\,s^{-1}$ in P_2. (B) Network response for different strengths (k_2) of P_2 with $k = 10^{-9}\,M/s$.

A



B



C

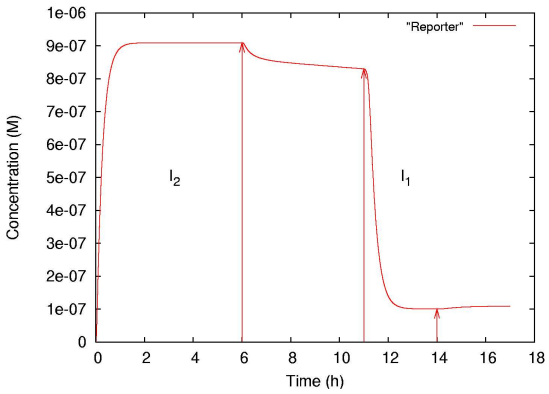


Figure 3.5: The bistable toggle switch. (A) Scheme: the network is made of two mutually repressing genes. Both transcription units contain an inducible promoter (P_1 and P_2) whose activity is regulated by a different kind of inducer (I_1 and I_2). The response of the network is estimated by calculating the concentration of a reporter protein (z) whose gene is connected to the second transcription unit. (B) Implementation. (C) Results of a simulation in four different steps: during the first six hours the circuit is fed with I_2; from the 6^{th} to the 11^{th} hour no inducer is sent to the network (I_2 has been removed); from the 11^{th} to the 14^{th} hour the circuit receives I_1; from the 14^{th} hour on I_1 is removed and the switch gets no more environmental signals.

Chapter 4

Part models

4.1 The two-operator promoter (promoter_$R_I R_I$)

In this section, the general model of the promoter used both in the seven-step cascade and in the repressilator is given. Cooperativity between the repressors is taken into account. O_1 has a higher affinity for the repressors than O_2. It is assumed that there is a unique value of γ, independent of the operator, which equals the value of λ.

Symbols

$\{O_1O_2\}_T$	total promoter concentration
α_1	association rate constant between repressors and O_1, no matter the state of O_2
β_1	dissociation rate constant between repressors and O_1, no matter the state of O_2
α_{2f}	association rate constant between repressors and O_2 when O_1 is free
β_{2f}	dissociation rate constant between repressors and O_2 when O_1 is free
α_{2t}	association rate constant between repressors and O_2 when O_1 is occupied
β_{2t}	dissociation rate constant between repressors and O_2 when O_1 is occupied
γ	association rate constant between n inducers and a repressor bound to an operator
λ	association rate constant between n inducers and a free repressor
μ	dissociation rate constant of n inducers from a free repressor
k_D	repressor constant decay rate
k_2^{lk}	transcription initiation frequency due to the leakage
R^a	free active repressors
R^i	free inactive repressors
I	free inducers
t_{cl}	clearance time

Reactions

$$O_1^t O_2^f \xrightarrow{k_D} O_1^f O_2^f \tag{4.1}$$

$$O_1^f O_2^t \xrightarrow{k_D} O_1^f O_2^f \tag{4.2}$$

$$O_1^t O_2^t \xrightarrow{k_D} O_1^f O_2^t \tag{4.3}$$

$$O_1^t O_2^t \xrightarrow{k_D} O_1^t O_2^f \tag{4.4}$$

$$R^a + O_1^f O_2^f \overset{(\alpha_1,\beta_1)}{\rightleftharpoons} O_1^t O_2^f \tag{4.5}$$

$$R^a + O_1^f O_2^t \overset{(\alpha_1,\beta_1)}{\rightleftharpoons} O_1^t O_2^t \tag{4.6}$$

$$R^a + O_1^t O_2^f \overset{(\alpha_{2t},\beta_{2t})}{\rightleftharpoons} O_1^t O_2^t \tag{4.7}$$

$$R^a + O_1^f O_2^f \overset{(\alpha_{2f},\beta_{2f})}{\rightleftharpoons} O_1^f O_2^t \tag{4.8}$$

$$Pol^{free} + O_1^f O_2^f \overset{(k_1,k_{-1})}{\rightleftharpoons} [PolO_1^f O_2^f] \xrightarrow{k_2} O_1^f O_2^f + Pol^{cl} \tag{4.9}$$

$$PoPS^{rt} \Longrightarrow [PolO_1^f O_2^f] \tag{4.10}$$

$$nI + R^a \overset{(\lambda,\mu)}{\rightleftharpoons} R^i \tag{4.11}$$

$$nI + O_1^t O_2^t \xrightarrow{\gamma} O_1^f O_2^t + R^i \tag{4.12}$$

$$nI + O_1^t O_2^t \xrightarrow{\gamma} O_1^t O_2^f + R^i \tag{4.13}$$

$$nI + O_1^t O_2^f \xrightarrow{\gamma} O_1^f O_2^f + R^i \tag{4.14}$$

$$nI + O_1^f O_2^t \xrightarrow{\gamma} O_1^f O_2^f + R^i \tag{4.15}$$

$$R^i \xrightarrow{k_D} nI \tag{4.16}$$

Equations

$$k_2 = {t_{cl}}^{-1} \tag{4.17}$$

$$O_1^f O_2^f = \{O_1 O_2\}_T - O_1^t O_2^f - O_1^f O_2^t - O_1^t O_2^t - [PolO_1^f O_2^f] \tag{4.18}$$

$$SiPS^b = n(-\lambda R^a I^n + \mu R^i) - n\gamma I^n (O_1^t O_2^f + O_1^f O_2^t + 2O_1^t O_2^t) + nk_D R^i \tag{4.19}$$

$$PoPS^b = -k_1 Pol^{free} O_1^f O_2^f + k_{-1}[PolO_1^f O_2^f] - PoPS^{lk} \tag{4.20}$$

$$\begin{aligned} FaPS^b = \quad & - \quad \alpha_1 R^a (O_1^f O_2^f + O_1^f O_2^t) + \beta_1 (O_1^t O_2^f + O_1^t O_2^t) + & (4.21) \\ & - \quad \alpha_{2f} R^a O_1^f O_2^f + \beta_{2f} O_1^f O_2^t - \alpha_{2t} R^a O_1^t O_2^f + \beta_{2t} O_1^t O_2^t & \\ & - \quad \lambda R^a I^n + \mu R^i & (4.22) \end{aligned}$$

$$PoPS^{lk} = k_2^{lk} (O_1^f O_2^t + O_1^t O_2^f + O_1^t O_2^t) \tag{4.23}$$

$$PoPS^{out} = k_2 [PolO_1 O_2] + PoPS^{lk} \tag{4.24}$$

$$\frac{d[PolO_1^f O_2^f]}{dt} = k_1 Pol^{free} O_1^f O_2^f - (k_{-1} + k_2)[PolO_1^f O_2^f] + PoPS^{rt} \tag{4.25}$$

$$\frac{dR^i}{dt} = \lambda R^a I^n - \mu R^i + \gamma I^n (O_1^t O_2^f + O_1^f O_2^t + 2O_1^t O_2^t) - k_D R^i \tag{4.26}$$

$$\begin{aligned} \frac{dO_1^f O_2^t}{dt} = \quad & - \quad \alpha_1 R^a O_1^f O_2^t + \beta_1 O_1^t O_2^t + \alpha_{2f} R^a O_1^f O_2^f - \beta_{2f} O_1^f O_2^t + & (4.27) \\ & - \quad \gamma O_1^f O_2^t I^n + \gamma O_1^t O_2^t I^n - k_D O_1^f O_2^t + k_D O_1^t O_2^t & \end{aligned}$$

$$\frac{dO_1^t O_2^f}{dt} = \alpha_1 R^a O_1^f O_2^f - \beta_1 O_1^t O_2^f - \alpha_{2t} R^a O_1^t O_2^f + \beta_{2t} O_1^t O_2^t + \\ - \gamma O_1^t O_2^f I^n + \gamma O_1^t O_2^t I^n - k_D O_1^t O_2^f + k_D O_1^t O_2^t \tag{4.28}$$

$$\frac{dO_1^t O_2^t}{dt} = \alpha_1 R^a O_1^f O_2^t - \beta_1 O_1^t O_2^t + \alpha_{2t} R^a O_1^t O_2^f - \beta_{2t} O_1^t O_2^t + \\ - 2\gamma O_1^t O_2^t I^n - 2k_D O_1^t O_2^t \tag{4.29}$$

4.2 sRNA

This part represents regions of the DNA that encode for small non-coding RNA (sRNA). These elements, in general, participate in the regulation of mRNA translation by interacting with the mRNA ribosome binding sites.
Different types of sRNA are known [Isaacs et al., 2006]. Our model is based on the crRNA-taRNA interaction [Isaacs et al., 2004] (and, consequently, on the *hok/sok* system [Franch et al., 1999]). Parameter values and concepts regarding sRNA degradation are taken from [Massé et al., 2003].
crRNA (cis-regulating RNA) and taRNA (trans-activating RNA) regulate the translation process with a *lock-key* mechanism. crRNA is the lock: it contains the complementary nucleotide sequence of the RBS that follows it on the DNA, so that they can interact and form a stem-loop as soon as the mRNA is transcribed. In this way the RBS is locked to the ribosomes. taRNA represents, on the contrary, a key. After being transcribed it is able to recognize the cognate crRNA, bind to it and open a stem loop allowing ribosomes to start protein synthesis.
In order to reproduce this model, we developed two new parts (crRNA and taRNA) and a new pool (sRNA pool, just for the taRNAs). A new signal carrier has to be considered (RNAPS, RNA Per Second). In the mathematical model (only to avoid confusion) this signal has been split into two different signals: $LoPS$ (Lock Per Second) and $KePS$ (Key Per Second).

4.3 The crRNA part

Even though this part is strictly connected to the adjacent RBS, it has to be modeled as a separate part, so that crRNAs are treated as external pieces of RNA which can bind to the RBS.
This part, furthermore, represents a pool of free locks available only to the adjacent RBS. Note that this RBS is different from the one described in the paper (see below).

Symbols

l^{free}	free locks (crRNAs)
$LoPS^b$	Locks Per Second ($LoPS$) received by the adjacent RBS
L	free lock sites on the DNA
$[PolL]$	polymerase + Lock complex
k_d	mRNA decay rate (lower than the taRNA's one)

Reactions

$$PoPS^{in} \Longrightarrow [PolL] \tag{4.30}$$

$$[PolL] \xrightarrow{k_{el}} Pol^{el} + L \tag{4.31}$$

$$l^{free} \xrightarrow{k_d} \tag{4.32}$$

Equations

$$k_{el} = \frac{v_{el}}{l_{crRNA}} \tag{4.33}$$

$$PoPS^{out} = k_{el}[PolL] \tag{4.34}$$

$$\frac{d[PolL]}{dt} = PoPS^{in} - k_{el}[PolL] \tag{4.35}$$

$$\frac{dl^{free}}{dt} = k_{el}[PolL] - k_d l^{free} + LoPS^b \tag{4.36}$$

4.4 The taRNA part

taRNAs (or keys) are produced by RNA polymerase (simple transcription). In order to have taRNAs in the system, this part has to be connected to a promoter and to a terminator. The effective taRNA production takes place in the terminator as soon as polymerases detach from the DNA, releasing its taRNA tail.

Symbols

K_e	free key sites on the DNA
$[PolK_e]$	polymerase + Key complex

Reactions

$$PoPS^{in} \Longrightarrow [PolK_e] \tag{4.37}$$

$$[PolK_e] \xrightarrow{k_{el}} Pol^{el} + K_e \tag{4.38}$$

Equations

$$k_{el} = \frac{v_{el}}{l_{taRNA}} \tag{4.39}$$

$$PoPS^{out} = k_{el}[PolK_e] \tag{4.40}$$

$$\frac{d[PolK_e]}{dt} = PoPS^{in} - k_{el}[PolK_e] \tag{4.41}$$

4.5 The terminator

This part is a slight modification of the terminator presented in the paper. It communicates with the sRNA (key) pool, which the free taRNAs enter after the end of the transcription process. It is supposed that all the taRNAs bound to readthrough polymerases are quickly degraded inside the next transcription unit, so that no readthrough signal affects the key production.

Symbols

$KePS^{out}$	Key Per Second ($KePS$) sent to the sRNA pool

Reactions

$$PoPS^{in} \Longrightarrow [PolT] \tag{4.42}$$

$$[PolT] \xrightarrow{\zeta} Pol^{free} + T \tag{4.43}$$

$$[PolT] \xrightarrow{\eta} Pol^{rt} + T \tag{4.44}$$

Equations

$$\frac{d[PolT]}{dt} = PoPS^{in} - (\zeta + \eta)[PolT] \tag{4.45}$$

$$PoPS^{out} = KePS^{out} = \zeta[PolT] \tag{4.46}$$

$$PoPS^{rt} = \eta[PolT] \tag{4.47}$$

4.6 The sRNA pool

This is the place where free taRNAs (keys) are stored. Keys come from a terminator and are exchanged with an RBS, where they interact with locks activating the translation process. According to [Massé et al., 2003], free keys decay at least three times more slowly than locks (which are attached to the mRNA). For sake of simplicity, in the following we will assume that this pool is connected to just one terminator and to one RBS.

Symbols

$KePS^{in}$	$KePS$ incoming from a terminator
$KePS^{b}$	$KePS$ incoming from an RBS
k_e^{free}	free taRNAs (keys)
k_{dk}	key decay rate ($k_{dk} < k_d$)

Reactions

$$KePS^{in} \Longrightarrow k_e^{free} \tag{4.48}$$

$$KePS^{b} \Longrightarrow k_e^{free} \tag{4.49}$$

$$k_e^{free} \xrightarrow{k_{dk}} \tag{4.50}$$

Equations

$$\frac{dk_e^{free}}{dt} = KePS^{in} + KePS^{b} - k_{dk}k_e^{free} \tag{4.51}$$

4.7 RBS

This version of the RBS is always connected to a crRNA which serves as a source of locks. As previously mentioned, the crRNA part plays the role of a lock pool for the RBS. Keys are also exchanged with their corresponding pool. Inside this RBS, locks can form a stem loop with the RBS itself or be silenced by the cognate keys.
Following [Franch et al., 1999], it is assumed that the reactions between locks and RBS, keys and stem-loop locks, keys and free locks take place with kinetic constants of the same order of magnitude. According to [Massé et al., 2003], the complex made of a key plus a lock (which is bound to the mRNA) decays faster than free keys, i. e., with the same decay rate (k_d) associated with the mRNA and the locks.

Symbols

$[lb]$	lock + mRNA-RBS complex
$[kl]$	key + lock complex
$LoPS^b$	$LoPS$ sent to the crRNA
$KePS^b$	$KePS$ sent to the sRNA pool
ϑ_l	$l^{free} + b$ association rate constant
ξ_l	$[lb]$ dissociation rate constant
ϑ_k	$k_e^{free} + [lb]$ association rate constant
ξ_k	dissociation rate constant of a key from a $[kl]$ complex; the products are $k_e^{free} + [lb]$
φ	$k^{free} + l^{free}$ association rate constant
χ	$[kl]$ dissociation rate constant

Reactions

$$PoPS^{in} \Longrightarrow [PolB] \overset{k_{el}^{RBS}}{\longrightarrow} Pol^{el} + B \tag{4.52}$$

$$PoPS^{rt} \Longrightarrow b \tag{4.53}$$

$$r^{free} + b \overset{(k_{1r},k_{-1r})}{\rightleftharpoons} [rb] \overset{k_{2r}}{\rightarrow} r^{cl} + b \tag{4.54}$$

$$b \overset{k_d}{\longrightarrow} \tag{4.55}$$

$$[rb] \overset{k_d}{\longrightarrow} r^{free} \tag{4.56}$$

$$[lb] \overset{k_d}{\longrightarrow} \tag{4.57}$$

$$l^{free} + b \overset{(\vartheta_l,\xi_l)}{\rightleftharpoons} [lb] \tag{4.58}$$

$$k_e^{free} + [lb] \overset{(\vartheta_k,\xi_k)}{\rightleftharpoons} [kl] + b \tag{4.59}$$

$$k_e^{free} + l^{free} \overset{(\varphi,\chi)}{\rightleftharpoons} [kl] \tag{4.60}$$

$$[kl] \overset{k_d}{\longrightarrow} \tag{4.61}$$

Equations

$$k_{el}^{RBS} = \frac{v_{el}}{l_{RBS}} \tag{4.62}$$

$$\frac{d[PolB]}{dt} = PoPS^{in} - k_{el}^{RBS}[PolB] \tag{4.63}$$

$$PoPS^{out} = k_{el}^{RBS}[PolB] \tag{4.64}$$

$$k_{2r} = (t_{cl}^r)^{-1} \tag{4.65}$$

$$\begin{aligned} \frac{db}{dt} &= k_{el}[PolB] - k_{1r}r^{free}b + (k_{-1r} + k_{2r})[rb] + \\ &- \vartheta_l l^{free}b + \xi_l[lb] + \vartheta_k k_e^{free}[lb] - \xi_k[kl]b + \\ &- k_d b + PoPS^{rt} \end{aligned} \tag{4.66}$$

$$\frac{d[rb]}{dt} = k_{1r}r^{free}b - (k_{-1r} + k_{2r} + k_d)[rb] \tag{4.67}$$

$$RiPS^b = -k_{1r} r^{free} b + (k_{-1r} + k_d)[rb] \tag{4.68}$$

$$RiPS^{out} = k_{2r}[rb] \tag{4.69}$$

$$\frac{d[lb]}{dt} = (\vartheta_l l^{free} + \xi_k[kl])b - (\xi_l + \vartheta_k k_e^{free} + k_d)[lb] \tag{4.70}$$

$$\frac{d[kl]}{dt} = \vartheta_k k_e^{free}[lb] - \xi_k[kl]b - k_d[kl] + \varphi k_e^{free} l^{free} - \chi[kl] \tag{4.71}$$

$$KePS^b = -\vartheta_k k_e^{free}[lb] + \xi_k[kl]b - \varphi k_e^{free} l^{free} + \chi[kl] \tag{4.72}$$

$$LoPS^b = -\vartheta_l l^{free} b + \xi_l[lb] - \varphi k_e^{free} l^{free} + \chi[kl] \tag{4.73}$$

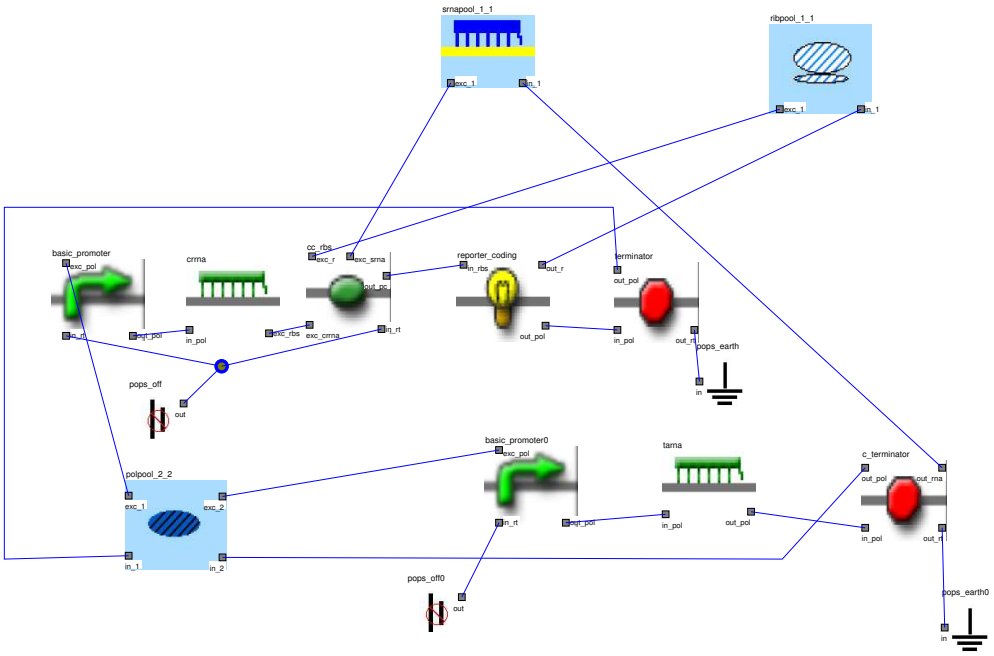


Figure 4.1: A simple circuit containing crRNA part, taRNA part and sRNA pool.

4.8 DNA

As stated in the Registry: *" DNA parts act as DNA itself. This includes restriction site complexes, DNA secondary structure, and spacers."* We considered the case where DNA encodes for spacers, in order to model in detail the functioning of a polycistronic mRNA (transcribed from an operon). When spacers are longer than 70 nucleotides, they have the only (small) effect of extending the time delay between the synthesis of two adjacent proteins. On the contrary, when they are quite small (here a standard length of 30 nucleotides has been considered) they can allow a readthrough effect in translation: some ribosomes do not leave the mRNA when they arrive at the protein coding stop codon but bind directly to the starting codon of the next gene, synthesizing a unique, big multi-protein without biological meaning (let us call it garbage).

We propose a simple model to estimate the production of every single "true" protein. To this end, we introduce the new part *spacer* and modify the RBS and protein coding parts as they are presented in the paper. The basic idea is that translation proceeds on two parallel tracks: the first occupied by the ribosomes that are synthesizing good proteins, the second by the readthrough ribosomes that come from the previous cistron and are synthesizing garbage. When ribosomes enter a spacer, some of them are released and go back to their pool freeing proteins or garbage. All other ribosomes, on the contrary, go on translating the next cistron. When they eventually leave the mRNA, they will release garbage.

4.9 The RBS

This RBS version allows the generation of a readthrough $RiPS$ signal ($RiPS^{rt}$) whereas the $PoPS^{rt}$ signal can be neglected. It can be preceded by a promoter or by a spacer.

Symbols

r^{rt}	readthrough ribosomes: they come from the previous cistron and carry a garbage tail;
$[r^{rt}b]$	readthrough ribosome + b complex
$RiPS_{in}^{rt}$	incoming readthrough $RiPS$
$RiPS_{out}^{rt}$	outcoming readthrough $RiPS$

Note that whereas r represents, as always, free ribosomes coming from the pool, r^{rt} represents readthrough ribosomes from the previous cistron.

Reactions

$$PoPS^{in} \Longrightarrow [PolB] \xrightarrow{k_{el}^{RBS}} Pol^{el} + B \tag{4.74}$$

$$r^{free} + b \overset{(k_{1r},k_{-1r})}{\rightleftharpoons} [rb] \xrightarrow{k_{2r}} r^{cl} + b \tag{4.75}$$

$$RiPS_{in}^{rt} \Longrightarrow [r^{rt}b] \xrightarrow{k_{2r}} r^{rt} + b \tag{4.76}$$

$$b \xrightarrow{k_d} \tag{4.77}$$

$$[rb] \xrightarrow{k_d} r^{free} \tag{4.78}$$

$$[r^{rt}b] \xrightarrow{k_d} r^{free} \tag{4.79}$$

Equations

$$k_{el}^{RBS} = \frac{v_{el}}{l_{RBS}} \tag{4.80}$$

$$\frac{d[PolB]}{dt} = PoPS^{in} - k_{el}^{RBS}[PolB] \tag{4.81}$$

$$PoPS^{out} = k_{el}^{RBS}[PolB] \tag{4.82}$$

$$k_{2r} = (t_{cl}^{r})^{-1} \tag{4.83}$$

$$\frac{db}{dt} = k_{el}^{RBS}[PolB] - k_{1r}r^{free}b + (k_{-1r} + k_{2r})[rb] + k_{2r}[r^{rt}b] - k_d b - RiPS_{in}^{rt} \tag{4.84}$$

$$\frac{d[rb]}{dt} = k_{1r}r^{free}b - (k_{-1r} + k_{2r} + k_d)[rb] \tag{4.85}$$

$$\frac{d[r^{rt}b]}{dt} = -(k_{2r} + k_d)[r^{rt}b] + RiPS_{in}^{rt} \tag{4.86}$$

$$RiPS^{b} = -k_{1r} r^{free} b + (k_{-1r} + k_d)[rb] + k_d[r^{rt}b] \tag{4.87}$$

$$RiPS^{out} = k_{2r}[rb] \tag{4.88}$$

$$RiPS_{out}^{rt} = k_{2r}[r^{rt}b] \tag{4.89}$$

4.10 The protein coding

This protein coding part differs from the original one: ribosomes are no more supposed to stop translating and to go back to their pool at the end of this part, but these actions are postponed into the spacer. The amount of synthesized proteins is also estimated inside the spacer even though it depends on the $RiPS$ signal generated here.

Symbols

a	AUG start codon on mRNA
$[r^{rt}a]$	readthrough ribosome + a complex
r^{el}	ribosomes in the elongation phase; they synthesize a protein
r_{el}^{rt}	readthrough ribosomes in the elongation phase; they synthesize garbage

Reactions

$$PoPS^{in} \Longrightarrow [PolA] \xrightarrow{k_{el}^{PC}} Pol^{el} + A \tag{4.90}$$

$$RiPS^{in} \Longrightarrow [ra] \xrightarrow{k_{el}^{r}} r^{el} + a \tag{4.91}$$

$$RiPS_{in}^{rt} \Longrightarrow [r^{rt}a] \xrightarrow{k_{el}^{r}} r_{el}^{rt} + a \tag{4.92}$$

Equations

$$k_{el}^{PC} = \frac{v_{el}}{l_{PC}} \tag{4.93}$$

$$\frac{d[PolA]}{dt} = PoPS^{in} - k_{el}^{PC}[PolA] \tag{4.94}$$

$$PoPS^{out} = k_{el}^{PC}[PolA] \tag{4.95}$$

$$k_{el}^{r} = \frac{v_{el}^{r}}{l_{PC}} \tag{4.96}$$

$$RiPS^{out} = k_{el}^{r}[ra] \tag{4.97}$$

$$\frac{d[ra]}{dt} = RiPS^{in} - k_{el}^{r}[ra] \tag{4.98}$$

$$\frac{d[r^{rt}a]}{dt} = RiPS_{in}^{rt} - k_{el}^{r}[r^{rt}a] \tag{4.99}$$

$$RiPS_{out}^{rt} = k_{el}^{r}[r^{rt}a] \tag{4.100}$$

4.11 The spacer

At the end of every single cistron, we have a spacer. Here, ribosomes are either released and sent back to their pool or, they continue translating (garbage) inside the next cistron. The total concentration of proteins released can be calculated by the spacer *efficiency* (e). This parameter depends only on the ribosomal dissociation (ζ^r) and readthrough (η^r) rate constants and permits to estimate how much of the $RiPS^{in}$ flux is converted into a final product (proteins). Note that the $FaPS^{out}$ flux has to be calculated only if the produced protein is a transcription factor. In this case, the spacer is also connected to a transcription factor pool.

Symbols

U	XXT stop codon on DNA
$[PolU]$	polymerase + XXT complex
$[ru]$	ribosome + u complex
$[r^{rt}u]$	readthrough ribosome + u complex
ζ^r	ribosomal dissociation rate constant
η^r	ribosomal readthrough rate constant
e	spacer efficiency

Reactions

$$PoPS^{in} \Longrightarrow [PolU] \xrightarrow{k_{el}} Pol^{el} + U \tag{4.101}$$

$$RiPS^{in} \Longrightarrow [ru] \tag{4.102}$$

$$[ru] \xrightarrow{\zeta^r} r^{free} + u \tag{4.103}$$

$$[ru] \xrightarrow{\eta^r} r^{rt} + u \tag{4.104}$$

$$RiPS^{rt}_{in} \Longrightarrow [r^{rt}u] \tag{4.105}$$

$$[r^{rt}u] \xrightarrow{\zeta^r} r^{free} + u \tag{4.106}$$

$$[r^{rt}u] \xrightarrow{\eta^r} r^{rt} + u \tag{4.107}$$

Equations

$$k_{el} = \frac{v_{el}}{l_{spacer}} \tag{4.108}$$

$$\frac{d[PolU]}{dt} = PoPS^{in} - k_{el}[PolU] \tag{4.109}$$

$$PoPS^{out} = k_{el}[PolU] \tag{4.110}$$

$$RiPS^{out} = \zeta^r([ru] + [r^{rt}u]) \tag{4.111}$$

$$RiPS^{rt}_{out} = \eta^r([ru] + [r^{rt}u]) \tag{4.112}$$

$$\frac{d[ru]}{dt} = RiPS^{in} - (\zeta^r + \eta^r)[ru] \tag{4.113}$$

$$\frac{d[r^{rt}u]}{dt} = RiPS^{rt}_{in} - (\zeta^r + \eta^r)[r^{rt}u] \tag{4.114}$$

$$e = \frac{\zeta^r}{\zeta^r + \eta^r} \tag{4.115}$$

$$\frac{dz}{dt} = eRiPS^{in} - k_D z \tag{4.116}$$

$$FaPS^{out} = eRiPS^{in} \tag{4.117}$$

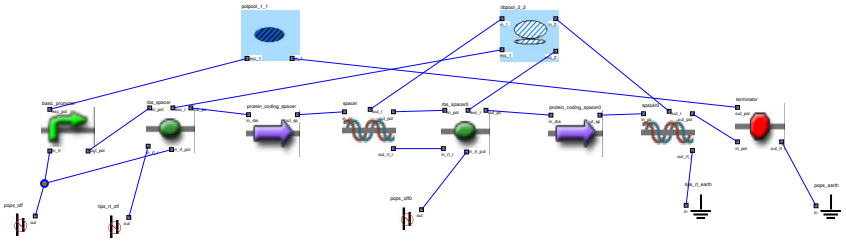


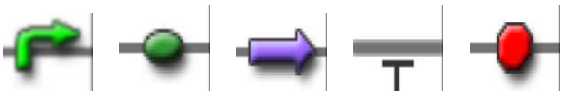
Figure 4.2: A simple circuit containing the spacer part.

4.12 Icons

A

B

C



D

E

Figure 4.3: Pools, plugs, parts and devices. From the left to the right: (A) polymerase, ribosome, transcription factor, signal and sRNA pool; (B) signal_off and earth plug; (C) promoter, RBS, protein coding, transcription factor coding and terminator; (D) RNA and spacer; (E) protein generator, reporter and composite device.

Bibliography

[Arkin et al., 1998] Arkin, A., Ross, J., and McAdams, H. H. (1998). Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected Escherichia coli cells. *Genetics*, 149(4):1633–1648.

[Bremer and Dennis, 1996] Bremer, H. and Dennis, P. P. (1996). Modulation of Chemical Composition and Other Parameters of the Cell by Growth Rate. In Neidhardt, F., III, R. C., Ingraham, J. L., Lin, E. C. C., Low, K. B., Magasanik, B., Reznikoff, W. S., Riley, M., Schaechter, M., and Umbarger, H. E., editors, *Escherichia coli and Salmonella typhimurium, Cellular and Molecular Microbiology*, pages 1553–1569. American Society for Microbiology, Whasington DC.

[Calogero et al., 1988] Calogero, R. A., Pon, C. L., Canonaco, M. A., and Gualerzi, C. O. (1988). Selection of the mrna translation initiation region by Escherichia coli ribosomes. *Proc Natl Acad Sci U S A*, 85(17):6427–6431.

[Elowitz and Leibler, 2000] Elowitz, M. B. and Leibler, S. (2000). A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338.

[Franch et al., 1999] Franch, T., Petersen, M., Wagner, E. G., Jacobsen, J. P., and Gerdes, K. (1999). Antisense RNA regulation in prokaryotes: rapid RNA/RNA interaction facilitated by a general U-turn loop structure. *J Mol Biol*, 294(5):1115–1125.

[Freistroffer et al., 1997] Freistroffer, D. V., Pavlov, M. Y., MacDougall, J., Buckingham, R. H., and Ehrenberg, M. (1997). Release factor RF3 in E.coli accelerates the dissociation of release factors RF1 and RF2 from the ribosome in a GTP-dependent manner. *EMBO J*, 16(13):4126–4133.

[Hooshangi et al., 2005] Hooshangi, S., Thiberge, S., and Weiss, R. (2005). Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc Natl Acad Sci U S A*, 102(10):3581–3586.

[Isaacs et al., 2006] Isaacs, F. J., Dwyer, D. J., and Collins, J. J. (2006). RNA synthetic biology. *Nat Biotechnol*, 24(5):545–554.

[Isaacs et al., 2004] Isaacs, F. J., Dwyer, D. J., Ding, C., Pervouchine, D. D., Cantor, C. R., and Collins, J. J. (2004). Engineered riboregulators enable post-transcriptional control of gene expression. *Nat Biotechnol*, 22(7):841–847.

[Kennell and Riezman, 1977] Kennell, D. and Riezman, H. (1977). Transcription and translation initiation frequencies of the Escherichia coli lac operon. *J Mol Biol*, 114(1):1–21.

[Knaus and Bujard, 1988] Knaus, R. and Bujard, H. (1988). PL of coliphage lambda: an alternative solution for an efficient promoter. *EMBO J*, 7(9):2919–2923.

[Lanzer and Bujard, 1988] Lanzer, M. and Bujard, H. (1988). Promoters largely determine the efficiency of repressor action. *Proc Natl Acad Sci U S A*, 85(23):8973–8977.

[Lewin, 2000] Lewin, B. (2000). *genes VII*. Oxford University Press, New York.

[Massé et al., 2003] Massé, E., Escorcia, F. E., and Gottesman, S. (2003). Coupled degradation of a small regulatory RNA and its mRNA targets in Escherichia coli. *Genes Dev*, 17(19):2374–2383.

[Tomsic et al., 2000] Tomsic, J., Vitali, L. A., Daviter, T., Savelsbergh, A., Spurio, R., Striebeck, P., Wintermeyer, W., Rodnina, M. V., and Gualerzi, C. O. (2000). Late events of translation initiation in bacteria: a kinetic analysis. *EMBO J*, 19(9):2127–2136.

[Tuttle et al., 2005] Tuttle, L. M., Salis, H., Tomshine, J., and Kaznessis, Y. N. (2005). Model-driven designs of an oscillating gene network. *Biophys J*, 89(6):3873–3883.