

# Combinatorial Gene Regulation Using Auto-Regulation

Rutger Hermsen<sup>1\*</sup>, Bas Ursem<sup>2</sup>, Pieter Rein ten Wolde<sup>2</sup>

**1** Center for Theoretical Biological Physics, University of California, San Diego, California, United States of America, **2** FOM Institute AMOLF, Amsterdam, The Netherlands

## Abstract

As many as 59% of the transcription factors in *Escherichia coli* regulate the transcription rate of their own genes. This suggests that auto-regulation has one or more important functions. Here, one possible function is studied. Often the transcription rate of an auto-regulator is also controlled by additional transcription factors. In these cases, the way the expression of the auto-regulator responds to changes in the concentrations of the “input” regulators (the response function) is obviously affected by the auto-regulation. We suggest that, conversely, auto-regulation may be used to optimize this response function. To test this hypothesis, we use an evolutionary algorithm and a chemical–physical model of transcription regulation to design model *cis*-regulatory constructs with predefined response functions. In these simulations, auto-regulation can evolve if this provides a functional benefit. When selecting for a series of elementary response functions—Boolean logic gates and linear responses—the *cis*-regulatory regions resulting from the simulations indeed often exploit auto-regulation. Surprisingly, the resulting constructs use auto-activation rather than auto-repression. Several design principles show up repeatedly in the simulation results. They demonstrate how auto-activation can be used to generate sharp, switch-like activation and repression circuits and how linearly decreasing response functions can be obtained. Auto-repression, on the other hand, resulted only when a high response speed or a suppression of intrinsic noise was also selected for. The results suggest that, while auto-repression may primarily be valuable to improve the dynamical properties of regulatory circuits, auto-activation is likely to evolve even when selection acts on the shape of response function only.

**Citation:** Hermsen R, Ursem B, ten Wolde PR (2010) Combinatorial Gene Regulation Using Auto-Regulation. *PLoS Comput Biol* 6(6): e1000813. doi:10.1371/journal.pcbi.1000813

**Editor:** Satoru Miyano, University of Tokyo, Japan

**Received:** October 26, 2009; **Accepted:** May 7, 2010; **Published:** June 10, 2010

**Copyright:** © 2010 Hermsen et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Funding:** This work is part of the research program of the “Stichting voor Fundamenteel Onderzoek der Materie (FOM)”, which is financially supported by the “Nederlandse organisatie voor Wetenschappelijk Onderzoek (NWO)”. Also, this work was supported by the Center for Theoretical Biological Physics (NSF PHY-0822283). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing Interests:** The authors have declared that no competing interests exist.

\* E-mail: hermsen@ctbp.ucsd.edu

## Introduction

Many transcription factors (TFs) in *Escherichia coli* regulate the transcription rate of their own gene. In fact, 59% of the TFs are known to auto-regulate and the list is growing [1,2]. Negative auto-regulation (auto-repression) occurs more frequently than positive auto-regulation (auto-activation), but both are very common: 71 auto-repressors and 34 auto-activators are found in the current databases (including 9 TFs that have binding sites for auto-activation as well as for auto-repression). This suggests that auto-regulation has one or several important functions [3,4]. In this paper, one possible function is explored. In general, the expression level of a gene is a function of the concentrations of the TFs that regulate its transcription rate. We propose that auto-regulation can naturally be used to optimize the shape of this response function.

Auto-regulating transcription factors are typically regulated by other TFs too. In fact, 23 auto-regulating TFs in *E. coli* are known to respond to at least *two* additional regulators [1,2]. In such cases, the response of the regulated TF to changes in the “input” TF concentrations must reflect an interplay between regulation and auto-regulation. Conversely, this suggests that auto-regulation could emerge as a result of natural selection on the shape of these responses.

In the past years, several other functions of auto-regulation have been proposed. Negative auto-regulation has been shown to

decrease the sensitivity of expression levels to intrinsic fluctuations in the transcription rate under certain conditions [5–7] and to mitigate variations due to changes in the bacterial growth rate [8]. In addition, auto-repression can speed up the response of expression levels after a sudden change in conditions [9,10]. In the presence of time delays, it can also create oscillations [11]. On the flip side, negative auto-regulation tends to reduce the sensitivity of the expression level to input signals [12,13]. The effects of positive auto-regulation are usually opposite to those of auto-repression: it slows down responses and tends to amplify intrinsic fluctuations. At first sight, these qualities may not seem very desirable. Yet, a slow response can be beneficial if a sensitive response to persisting signals is desired while fast fluctuations in the input signal should be ignored [12]. Occasionally the fact that auto-activation can provide bi-stability may also be useful [14]. Each of these qualities could be relevant in some cases; our new suggestion does not contradict or replace any of them.

To study the benefits of auto-regulation we use a computational approach that we developed recently [15]. In this approach, an evolutionary algorithm and a physical–chemical model of transcription regulation are integrated to design *in silico cis*-regulatory regions with predefined response functions. The evolutionary algorithm subjects a population of model *cis*-regulatory regions to rounds of mutation and selection. The mutations are introduced at the level of base-pair sequences while the selection step is based on the emerging network properties

## Author Summary

Bacteria adjust which proteins they make, and how many copies of each kind, depending on their environment. The production rate of each regulated protein is controlled by a special class of proteins called transcription factors. The rate at which a certain protein is produced usually depends on the cellular concentrations of a few such transcription factors. When circumstances change, the concentrations of these transcription factors alter too and consequently the production rates of all proteins regulated by them are adjusted. Interestingly, many transcription factors also regulate their own synthesis rate. This suggests that this self-regulation must have one or more important functions. In this article we study one possible function. In order for cells to function properly each protein concentration has to respond in a particular way to changes in transcription factor concentrations. We have studied how bacteria can optimize and fine-tune these responses. To this end, we formulated a physical model of the regulation by transcription factors and performed computer simulations. These simulations show that self-regulation—and in particular self-activation—is often a useful tool to achieve the prescribed response. Therefore we conclude that natural selection on the regulation of protein levels could naturally lead to self-regulation.

calculated using the model of transcription regulation. In the course of these simulations complex promoter designs develop that perform the desired function; these designs often reveal new design principles. In earlier work, auto-regulation was not included in this method. In contrast, we now use an extended version of the method to design *cis*-regulatory constructs that *can* exploit feedback.

Many *cis*-regulatory regions in real cells essentially implement logical decisions [15–17]. We therefore study the class of response functions that can be interpreted as analogue equivalents of logic gates. *Gates* are computational devices that produce an output signal depending on one or more input signals; *logic gates* are gates that implement a binary (Boolean) decision rule. For example, a transcriptional AND gate would be a gene whose expression (the “output”) is regulated by two TFs (the “inputs”, TF1 and TF2) such that it is transcribed only if both TF1 *and* TF2 have a sufficient expression level [16]. We refer to Table 1 for the definitions of other logic gates. Even though it has proven fruitful

to think of promoters as analog approximations of logic gates, we stress that gene expression levels are of course not actually binary and that we do not treat them as binary in the models below.

We analyze the *cis*-regulatory sequences resulting from the simulations by calculating DNA footprints for the resulting transcription factors and promoter sequences. These footprints show that auto-regulation—in particular auto-*activation*—is often used in these *cis*-regulatory regions; indeed, further analysis shows that auto-activation can be used to construct “better” transcriptional logic gates by allowing for more switch-like, “steep” response functions. However, the use of auto-regulation in shaping response functions is not limited to creating switch-like functions. To demonstrate this, we also applied our method to the design of *cis*-regulatory constructs that respond in a *linear* fashion to input concentrations. Again we find that auto-activation emerges spontaneously in the results.

Finally, we also performed simulations in which we selected for designs with desirable dynamical qualities. First, we adjusted the method to select for gates with a short response time. Second, we performed selection against intrinsic fluctuations. In agreement with earlier results [5–7,9,10], auto-repression evolved in both cases, demonstrating how auto-repression can be used to speed up response times or to reduce intrinsic fluctuations.

Before describing the results we first provide a detailed description of the model and the algorithm used.

## Methods

We combine a model of transcription regulation and an evolutionary algorithm to design *in silico cis*-regulatory regions with a predefined function. The model of transcription regulation is constructed such that all properties of a model regulatory network follow entirely from the sequences of TFs and *cis*-regulatory regions. Binding sites of TFs are therefore not specified beforehand but appear gradually in the course of the simulations. The model is an extension to the one described in detail in our earlier publication [15]. The main innovation is that auto-regulatory interactions are now also included, so that auto-regulation can evolve if this is beneficial.

## Model of transcriptional regulation

We consider one “output” gene, *tf3*, and at most two “input” transcription factors, TF1 and TF2. The gene *tf3* codes for another transcription factor called TF3. All three TFs can regulate the

**Table 1.** Truth tables of transcriptional logic gates.

input TFs		output ( $c_3^*$ )					
$c_1$	$c_2$	AND	OR	NOR	NAND	ACT <sup>†</sup>	IN <sup>†</sup>
low	low	low	low	high	high	low	high
low	high	low	high	low	high	low	high
high	low	low	high	low	high	high	low
high	high	high	high	low	low	high	low

<sup>†</sup>The ACT and INH gates have only one input; they depend on  $c_1$  only.

Logic gates are devices that perform elementary binary computations, mapping multiple input signals to one output signal. Here we consider *transcriptional* logic gates. Transcription systems are analogue systems and therefore never truly binary, but it is often useful to think of them as continuous approximations of logic gates. Here, we consider logic gates with one or two inputs (the concentrations  $c_1$  and  $c_2$  of two transcription factors), and one output (the steady-state expression level of the regulated gene:  $c_3^*$ ). The table specifies, for the six logic gates used in our study, the value of the appropriate output  $c_3^*$  (“low” or “high”) for all input concentrations  $c_1$  and  $c_2$  (“low” or “high”). All gates are defined for input concentrations in the domain  $[0, 1\mu\text{M}]$  only; concentrations above 500nM are considered high and those below 500nM are low. The acronyms of the gates summarize their function; for instance, the output of an AND gate should be high only when both  $c_1$  *and*  $c_2$  are high. Note that that by definition the ACT (activate) and IN (inhibit) gates have only one input,  $c_1$ ; indeed, their output does not depend on  $c_2$ .

doi:10.1371/journal.pcbi.1000813.t001

transcription rate of  $tf\beta$  by binding to its *cis*-regulatory region. (See Fig. 1 for an illustration of the model.)

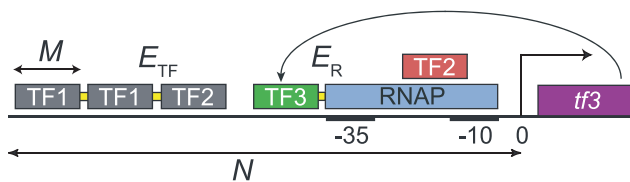
The *cis*-regulatory region and the TFs are represented as nucleotide sequences and amino-acid sequences respectively. All TFs can bind anywhere on the *cis*-regulatory region, but the affinity of a TF for a particular site depends on the sequences of the TF and the site. For our purpose, it is sufficient to only model the DNA-binding domains of the TFs explicitly. We assume that  $M$  amino-acids in these domains are responsible for the DNA-binding specificity and therefore represent each TF as an amino-acid sequence of length  $M$ . We choose  $M = 10$  in our simulations because known binding sites in *E. coli* typically have length 6 to 15 and usually one base pair interacts with  $\approx 1$  amino acid in TF-DNA binding [2]. The *cis*-regulatory region of  $tf\beta$  is a base-pair sequence of length  $N$ ; we take  $N = 100$  because in *E. coli* most transcription factors bind within  $\approx 100$ bp from the start of transcription [18]. By the rules specified below all interactions between TFs, RNA polymerase (RNAP) and the *cis*-regulatory region can be deduced from these sequences; therefore each transcriptional gate is completely specified by them.

The various molecules interact in the following ways:

**TF-DNA interactions.** Each TF  $a$  can bind to any site  $O$  on the *cis*-regulatory region, but the affinity of  $a$  for  $O$  depends on the DNA sequence of  $O$  and the amino-acid sequence of  $a$ . Each amino acid interacts with exactly one base pair, and the total binding free energy  $E_{a,O}$  is the sum of the contributions of each amino-acid-base-pair contact. This means that the binding free energy of a TF with amino acids  $a_i$  to a binding site with base pairs  $O_i$  is given by

$$E_{a,O} = \sum_{i=1}^M U_{a_i, O_i}. \quad (1)$$

Here  $U_{i\mu}$  is a  $20 \times 4$  matrix containing the binding free energies associated with each amino-acid-base-pair contact. We used the



**Figure 1. Illustration of the model of transcription regulation.**

The model describes the transcriptional regulation of a gene  $tf\beta$  by two transcription factors, TF1 and TF2. In addition, auto-regulation is included: the gene product TF3 of  $tf\beta$  can regulate the transcription of  $tf\beta$ . TFs act by binding to  $tf\beta$ 's *cis*-regulatory region, represented as a string of  $N = 100$ bp located directly upstream of the start of transcription. The TF binding domains count  $M = 10$  amino acids, and bind to binding sites of length  $M$  bp. TFs can bind anywhere on the *cis*-regulatory region but with varying affinity determined by the DNA sequence. When two TFs bind within a distance less than  $d = 4$ bp, they interact with energy  $E_{TF} = 3.40k_B T$ , as is indicated schematically by a yellow connection between the TFs. This way cooperative binding is included. The core promoter, consisting of the  $-10$  and  $-35$  hexamers, is marked; when RNA polymerase (RNAP) binds to it, it blocks both hexamers and the spacer between them. When a TF binds close to the RNAP we assume an interaction energy  $E_R = 3.40k_B T$ ; thus the mechanism of regulated recruitment is included. The TF that binds to a site overlapping with the core promoter is red to indicate that it represses transcription by steric hindrance; the green TF is an activator, as it recruits RNAP from its binding site. The gray TFs bind too far upstream to influence the transcription rate. doi:10.1371/journal.pcbi.1000813.g001

matrix given in [19], based on crystallographically solved protein-DNA complexes.

**TF-TF interactions.** TFs are assumed to mutually interact in two ways. First, TFs cannot bind simultaneously to overlapping binding sites due to steric hindrance. This introduces competition for binding to overlapping binding sites. Second, if two TFs bind within a distance of  $d = 4$  base pairs, they interact cooperatively: if one of the TFs is bound, the equilibrium dissociation constant of the other TF is decreased by a factor  $\omega = 30$ , corresponding to an interaction energy  $E_{TF} = \ln(\omega)k_B T = 3.40k_B T$ . In reality,  $E_{TF}$  is typically 1 to  $6k_B T$  (about 1 to  $4kcal/mol$ ), leading to  $\omega \approx 10 - 100$  [20]. Many TFs interact by direct contact, but indirect interactions may also occur, for instance if one TF bends, stretches or super-coils the local DNA and thus increases the affinity of the other TF. In the context of this model it is irrelevant which mechanism is responsible for the cooperativity.

**RNAP-DNA interactions.** RNAP is assumed to bind only to the basal promoter. The affinity of the RNAP- $\sigma$ -complex for the core promoter is determined by the  $-10$  and  $-35$  consensus hexamer sequences on the *cis*-regulatory region of  $tf\beta$ . We determine the binding free energy of RNAP- $\sigma$  for a core promoter by comparing the  $-10$  and  $-35$  hexamers to a large set of real *E. coli* promoters, taken from reference [21]. To every base pair  $p_i$  at position  $i$  within the  $-10$  and  $-35$  hexamers we assign a score  $s_i$ ; it equals the fraction of real *E. coli* promoters that have  $p_i$  at that particular position, normalized by the random fraction  $1/4$ . Next, the binding energy  $E_p$  of RNAP to that particular core promoter can be estimated by [22–24]:

$$E_p = k_B T \sum_i \log(s_i). \quad (2)$$

Thus, promoter sequences that are more similar to the consensus sequence of real promoters yield a higher affinity.

**TF-RNAP interactions.** TFs interact with RNAP and influence the transcription rate by the principles of regulated recruitment [25]. If TFs bind to a site overlapping the basal promoter they block the binding of RNAP and thus repress transcription. On the other hand, if they bind within a distance  $d = 4$  from the promoter we assume that the TF and RNAP interact cooperatively. If the TF is bound to its binding site, this decreases the dissociation constant of RNAP binding to the promoter by a factor  $\omega = 30$  (corresponding to an interaction energy  $E_R = 3.40k_B T$ ); thus, the TF recruits RNAP and activates transcription.

The transcription rate of  $tf\beta$  is assumed to be proportional to the equilibrium fraction of time the promoter is occupied by RNAP [20]. This occupancy can be computed given all sequences and the concentrations of the TFs (denoted by  $c_1$ ,  $c_2$  and  $c_3$ ). For this we use the statistical mechanics formalism originally developed in Ref. [20] and explained in detail in references [26] and [15]. Briefly, the method calculates the partition sum  $Z_{on}(c_1, c_2, c_3)$  of all states of the system in which RNAP is bound to the promoter, and the partition sum  $Z_{off}(c_1, c_2, c_3)$  of all states in which RNAP is *not* bound; the fractional occupancy of the promoter is then given by:

$$p_{on}(c_1, c_2, c_3) = \frac{Z_{on}(c_1, c_2, c_3)}{Z_{on}(c_1, c_2, c_3) + Z_{off}(c_1, c_2, c_3)}. \quad (3)$$

Note that, in general,  $p_{on}$  is a function of all three TF concentrations. To efficiently take into account all possible configurations in which TFs could be bound to the *cis*-regulatory region we use the efficient recursive (dynamic programming) algorithm presented in [15].

The model described above contains the generic mechanisms that are responsible for many of the known transcription regulatory interactions. Yet, several mechanisms that play a role in specific systems are not included; for instance, the model does not include DNA looping, extended  $-10$  regions or twisting or bending of the DNA by TFs. This suggests that real transcriptional regulatory systems are probably more flexible than our model systems. However, by endowing our model with only the basic features known to be important in virtually any transcription system we ensure that the resulting designs do not rely on exotic mechanisms.

### Model of the dynamics

We model the dynamics of the concentration of TF3,  $c_3(t)$ , by the following ordinary differential equation:

$$\frac{dc_3(t)}{dt} = \alpha p_{\text{on}}(c_1, c_2, c_3(t)) - \beta c_3(t). \quad (4)$$

Here  $\alpha$  is the maximal production rate of TF3, and  $\beta$  is the degradation rate constant of TF3. The function  $p_{\text{on}}(c_1, c_2, c_3)$  was defined above. In this simplified description, transcription and translation are concatenated and translational regulation is not included.

The concentrations  $c_1$  and  $c_2$  of TF1 and TF2 are considered the inputs of the gate. Assuming that the system is mono-stable (bi-stability is discussed below) equation 4 defines a unique steady state for each set of input concentrations  $(c_1, c_2)$  in which  $c_3$  has a value  $c_3^*(c_1, c_2)$ . This steady-state concentration is considered the output of the gate. Because time delays between transcription initiation and translation are ignored, oscillations are excluded and  $c_3^*(c_1, c_2)$  can be calculated by propagating the dynamics numerically from any initial condition until the steady state is reached. (If a gate has only one input, the dependence on  $c_2$  is simply dropped.)

We choose the constants  $\alpha$  and  $\beta$  such that  $\alpha/\beta = 1\mu\text{M}$ ; this ensures that  $c_3^*$  stays within the range  $0\mu\text{M} - 1\mu\text{M}$ . Apart from this ratio the values of  $\alpha$  and  $\beta$  are irrelevant because in this work we are not interested in absolute time scales of the dynamics; for simplicity we use a time unit  $\tau$  such that  $\beta\tau = 1$ .

### Evolutionary algorithm

In order to design networks with a prescribed function an evolutionary algorithm was used. A population of 200 transcriptional gates was subjected to 1000 cycles of mutation, selection and replication. Initially, all gates had random sequences. Auto-regulation was not imposed, but the system was free to exploit it by evolving binding sites for TF3. The details of the evolutionary algorithm were chosen to combine an effective optimization of the gates with computational efficiency; we emphasize that we do not intend to faithfully mimic biological evolution.

Several types of mutations were included. First, a base substitution could occur in *cis*-regulatory sequences (with probability  $P_{\text{bp}} = 0.28$  per *cis*-regulatory region). If this happened, a base pair was selected at random from the *cis*-regulatory sequence and substituted by a randomly chosen nucleotide. Second, insertions or deletions of a random base pair occurred in *cis*-regulatory regions (with probability  $P_{\text{indel}} = 0.56$ ). Third, we applied point mutations to the sequences of the TFs (with probability  $P_{\text{aa}} = 0.30$  per TF), in which case one randomly chosen amino acid in the sequence was replaced by a random alternative. The exact mutation rates are not crucial for the results, as long as the rates are (i) high enough to generate significant variation and (ii) low enough to allow high-quality gates to persist in the population.

For the selection step a fitness score  $F_{\text{RF}} \equiv C - \gamma_{\text{RF}} M_{\text{RF}}$  was used. Here  $M_{\text{RF}}$  (where RF stands for Response Function) measures the deviation of the response function from a predefined goal function (the desired response function). It was computed as follows. When evaluating gates with two inputs,  $c_1$  and  $c_2$ , the output level  $c_3^*(c_1, c_2)$  was computed for 16 combinations of the input concentrations:  $c_1, c_2 \in \{0\text{nM}, 333\text{nM}, 667\text{nM}, 1000\text{nM}\}$  (see the red dots in Fig. 2B). Next, the differences between these output levels and the goal function were computed.  $M_{\text{RF}}$  was defined as the sum of the squares of these deviations. If the gate had only one input, the definition was analogous, except that seven input values were used, equally spaced in the interval  $[0\text{nM}, 1000\text{nM}]$ . The constant  $\gamma_{\text{RF}}$  is required to make  $F_{\text{RF}}$  dimensionless (for simplicity,  $\gamma_{\text{RF}} = 1\mu\text{M}^{-2}$ ) and  $C$  is an arbitrary constant large enough to ensure  $F_{\text{RF}} > 0$ . Based on the fitness scores, the top 20% of the population were selected and the remaining gates discarded. Subsequently the population was brought back to its initial size by duplicating gates randomly chosen from the survivors of the selection process.

If auto-activation evolved, the system could become bi-stable. In bi-stable systems, given the inputs  $(c_1, c_2)$  two different values of  $c_3$  are stable under the dynamics of the system (equation 4) so that the output concentration  $c_3^*$  is not uniquely defined by the input concentrations. Even though bi-stability is likely to occur in some real transcription networks we decided that such systems do not qualify as gates, since gates by definition map input states to a uniquely defined output state. The fitness function therefore contained an additional term that was designed to penalize bi-stability. When evaluating the fitness of a gate, we always computed the steady-state value  $c_3^*$  twice for  $12 \times 12$  input values  $(c_1, c_2)$ : once by propagating the differential equation 4 using initial condition  $c_3(0) = 0\mu\text{M}$  and once using  $c_3(0) = 1\mu\text{M}$ . If the results were different, the difference squared was added to the fitness function, which was sufficient to assure that the particular gate was eliminated by the selection process. However, because this method did not exclude bi-stability for all possible input values we also checked afterward whether the results were bi-stable.

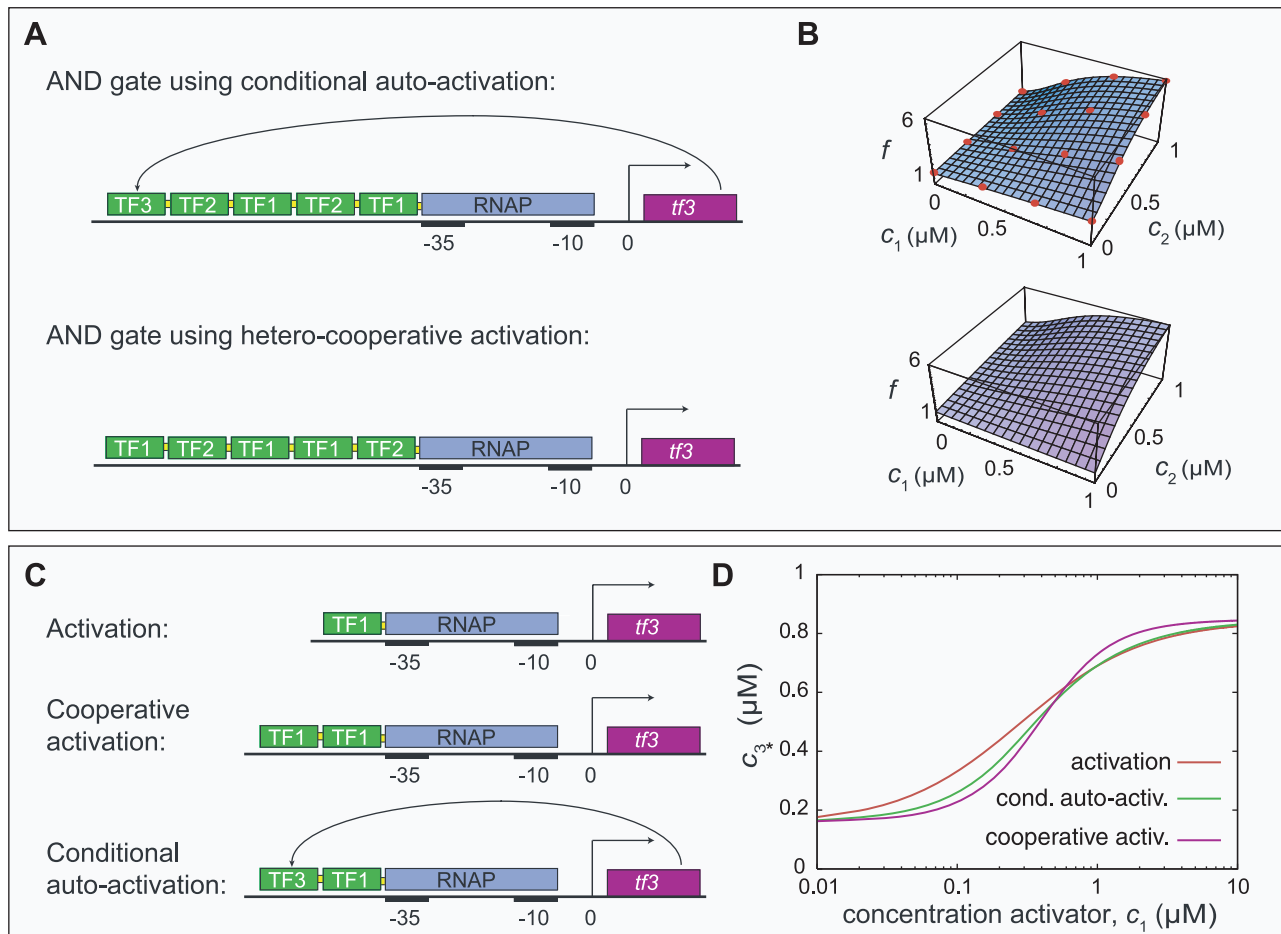
### Definition of the gates

All gates are defined for input concentrations in the domain  $[0\mu\text{M}, 1\mu\text{M}]$  only. The logic gates are specified in Table 1. In addition we define LACT, LIN, MEAN and NMEAN gates. A LACT (linear activate) gate has one input,  $c_1$ , and the output  $c_3^*(c_1)$  responds as  $c_3^*(c_1) = c_1$ . A LIN (linear inhibit) gate also has one input, but responds according to  $c_3^*(c_1) = 1\mu\text{M} - c_1$ . A MEAN gate is linear in two inputs,  $c_1$  and  $c_2$ , and obeys  $c_3^*(c_1, c_2) = (c_1 + c_2)/2$ . Lastly, we define NMEAN to have the following linearly decreasing response function:  $c_3^*(c_1, c_2) = 1\mu\text{M} - (c_1 + c_2)/2$ .

We repeated the simulations for each of the gates 20 times with different random seeds.

### Quantifying the degree of auto-regulation

In order to quantify the importance of auto-regulation in a particular design we defined the measure  $M_{\text{AR}}$  (referred to as the “feedback measure”). First, we calculated the response function  $c_3^*(c_1, c_2)$  for the particular design. Then we artificially removed all possible binding sites for TF3 by setting the affinity of TF3 for all sites on the *cis*-regulatory region to zero and calculated the response function again; we call the result  $c_m^*(c_1, c_2)$ . In the absence of auto-regulation one should find  $c_m^*(c_1, c_2) = c_3^*(c_1, c_2)$ , but if auto-regulation does play a role the two functions differ. Therefore the difference between these functions is a measure of the degree of auto-regulation; we define  $M_{\text{AR}}$  as the mean of the



**Figure 2. Conditional auto-activation.** In some simulation results, auto-activation occurs only in the presence of another transcription factor (TF). We call this conditional auto-activation. The figure presents two examples. Fig. A: The promoter of an AND gate using conditional auto-activation and, for comparison, one using hetero-cooperative activation. Both designs emerged in the simulations. In the first case, the regulated gene *tf3* codes for a transcription factor TF3 that binds to its own *cis*-regulatory region. However, from its binding site, TF3 can activate transcription only indirectly, by facilitating the binding of TF1 and TF2 to their binding sites. As a result, the auto-activation depends on the presence of TF1 and TF2 ( $c_1$  and  $c_2$ ). Fig. B shows plots of the expression level of *tf3* (fold-change  $f$  vs. the concentrations  $c_1$  and  $c_2$  of TF1 and TF2) resulting from the *cis*-regulatory regions in Fig. A. The red dots show the values of  $(c_1, c_2)$  that were used to evaluate the fitness of the gate (see Methods). Fig. C and D show the same mechanism in a simplified model inspired by the simulation results. Plot D compares the response functions corresponding to three activation systems depicted in C. In all cases, a single TF activates the expression of a gene *tf3* coding for another transcription factor, TF3. The first two scenarios constitute conventional activation systems with one or two binding sites. In the third scenario, one binding site is replaced by an operator for TF3, which introduces a positive feedback loop depending on the presence TF1. The binding affinities of all sites are optimized using the fitness function described in the main text. The response of the conditional auto-activation system is clearly sharper/more sensitive than the one using a single activation site. Cooperative auto-activation by two sites, however, leads to a slightly sharper response. The results suggest that conditional auto-activation is an alternative design principle that can be used to sharpen responses.  
doi:10.1371/journal.pcbi.1000813.g002

squared differences  $c_m^*(c_1, c_2) - c_3^*(c_1, c_2)$  over 16 combinations of the input concentrations  $c_1, c_2 \in \{0\text{nM}, 333\text{nM}, 667\text{nM}, 1000\text{nM}\}$  (again, the red dots in Fig. 2). If auto-regulation is not being exploited by a certain design  $M_{AR}$  is generally small ( $< 1\text{nM}^2$ ). If, on the other hand, auto-regulation is used the resulting value is typically in the range  $10^3 - 10^6\text{nM}^2$ . (For instance, if all 16 points shift by 100nM when auto-regulation is removed  $M_{AR} = 10^4\text{nM}^2$ .)

### Recognizing binding sites

During the simulations, binding sites emerge in the initially random *cis*-regulatory promoter sequences. However, since the equilibrium binding constants have continuous values there is no fundamental distinction between binding sites and non-binding sites. Recognizing binding sites is further complicated by the fact that, in particular in the presence of cooperativity, weak binding

sites can be important. Nevertheless, in order to understand the design principles of a particular gate we wish to identify which binding sites are necessary and sufficient to explain the observed promoter response. This problem is not an artifact of our models: the exact same conceptual problems occur whenever one tries to identify the binding sites of real TFs by experiment.

Since a direct cutoff in terms of the equilibrium constants would eliminate possibly important weak sites we use computational “DNA footprints” (analogous to experimental techniques such as DNase I footprinting) to select those sites that are likely to be important. For each TF  $a$  and each site  $O$ , we calculate the steady-state occupancy  $o_{a,O}(c_1, c_2)$  for four sets of input concentrations  $(c_1, c_2) \in \{(0, 0), (0, 1\mu\text{M}), (1\mu\text{M}, 0), (1\mu\text{M}, 1\mu\text{M})\}$ . Sites that influence the response of the gate should have a significant occupancy in at least one of these digital footprints.

We define  $o_{a,O}^{\max}$  to be the maximal occupancy of site  $O$  by TF  $a$  over the four conditions. Figure 4 in Text S1 shows a histogram of these occupancies for all TFs and all sites using data gathered from the results of 200 simulations. This histogram is bi-modal. The vast majority of the maximal occupancies  $o_{a,O}^{\max}$  have negligible values but a second peak occurs at  $o_{a,O}^{\max} \approx 0.9$ . This peak is the result of selection pressure and is associated with functional binding sites. Based on this histogram, we use a rather stringent cut-off at  $o_{a,O}^{\max} = 0.3$  to separate binding sites from pseudo binding sites. Simplified models that only take into account these selected binding sites and assume that all other binding affinities are zero usually accurately reproduce the response function of the full, unsimplified system. In rare cases where this is not the case the threshold can be lowered to obtain more accurate but more complex models; this was not necessary for the examples presented below. (See the Text S1 for more details and examples of footprinting profiles.)

## Results

Below, we describe the results of the simulations. We regularly compare the results to our previous work in which auto-regulation was excluded (Ref. [15]). In some cases the simulations presented here resulted in the same designs as before. But in other cases the designs exploited auto-regulation to arrive at novel and often superior designs. We identified several mechanisms that are used repeatedly in the results and present them below. Details about several of the analyses below can be found in Text S1.

### Conditional auto-activation leads to steep responses

The first mechanism that our scheme elucidated, we called *conditional auto-activation*. This mechanism occurred in AND and ACT (activation) gates (see Table 1 for the definitions), in which cooperative activation plays a key role. In those gates, conditional auto-activation is used to create a steep, switch-like response. As an example, we first discuss the design of AND gates.

In simulations in which auto-regulation was excluded by the method, the resulting AND gate designs always consist of a tandem array of binding sites to which TF1 and TF2 bind cooperatively (see Fig. 2A) [15,16]. We called this a hetero-cooperative module. This design functions as follows. Crucially, the binding site from which RNAP is recruited (the site directly next to the core promoter) is too weak to considerably activate transcription on its own. As a result, only when TF1 and TF2 are both present at sufficient concentrations they bind cooperatively and activate transcription, as the definition of an AND gate requires. In the new simulations, in which auto-regulation can evolve, this design still emerged in 14 out of 20 simulation runs. Each of these gates has a feedback measure  $M_{AR} < 10^{-7} \text{nM}^2$ , proving that auto-regulation does not play any role.

The remaining 6 simulation runs resulted in conditional auto-activation. In these gates the feedback measure  $M_{AR}$  was high, in the range  $2.0 \times 10^3 - 1.5 \times 10^4 \text{nM}^2$ . The new design looks very similar to the old one (see Fig. 2A and B). However, the hetero-cooperative module now also contains a binding site for TF3, which leads to a positive feedback loop. Importantly, TF3 bound at its binding site cannot recruit RNAP directly; instead, it interacts with the hetero-cooperative activation module. As a result, the auto-activation is conditional on the presence of TF1 and TF2. As the concentrations of TF1 and/or TF2 increase, the auto-activation is gradually turned on, leading to a sudden (steep, switch-like) response.

The exact same mechanism is exploited by some ACT gates. Out of the 20 simulations of ACT gates, 3 resulted in conditional

auto-activation ( $M_{AR}$  values were  $3.6 \times 10^2 \text{nM}^2$ ,  $3.4 \times 10^3 \text{nM}^2$  and  $1.6 \times 10^4 \text{nM}^2$ ), while the other 17 did not use auto-regulation ( $M_{AR} < 10^{-7} \text{nM}^2$ ).

The basic mechanism can be studied in minimal models inspired by the simulation results. In Fig. 2C and D, we compare three activation mechanisms. The first scenario is conventional activation by a single TF1 binding site. In the second scenario only a homo-cooperative activation module is present, consisting of two binding sites for TF1. In the third scenario, the auxiliary TF1 site is replaced by a binding site for TF3, introducing conditional auto-activation. In all designs we chose the binding site affinities such that they maximize the fitness function for the ACT gate. Conditional auto-activation indeed produces a response that is steeper than the one resulting from the design with a single activator binding site (Fig. 2D). However, the conventional cooperative design with two binding sites gives an even steeper result. The results imply that, after one binding site has evolved for the activator TF1, the response can be improved in two ways: by adding an additional site for TF1 (leading to cooperative activation) or by adding a site for TF3 (resulting in conditional auto-activation). Which design emerges therefore depends critically on the actual sequences and mutations occurring in the population. This explains why cooperative activation and conditional auto-activation show up as alternatives in the simulations.

The effect of conditional auto-activation can be understood quantitatively by studying the minimal model mathematically. The response function  $c_3^*(c_1)$  for the minimal model follows from the condition  $\alpha p_{\text{on}}(c_1, c_3^*) = \beta c_3^*$  and is given by

$$\frac{c_3^*(c_1)}{k^*} = \frac{-B(c_1) + \sqrt{B(c_1)^2 + 4A(c_1)C(c_1)}}{2A(c_1)}, \quad (5)$$

with

$$A(c_1) \equiv r(q_p(1 + \omega^2 c_1/k) + 1 + \omega c_1/k),$$

$$B(c_1) \equiv r(q_p(1 + \omega c_1/k) + 1 + c_1) - q_p(1 + \omega^2 c_1/k),$$

$$C(c_1) \equiv q_p(1 + \omega c_1/k).$$

Here  $k$  and  $k^*$  denote the dissociation constants for TF1 and TF3 binding to their respective operator,  $q_p$  is the concentration of RNAP normalized by the dissociation constant of RNAP binding to the promoter, and  $r \equiv \beta k^*/\alpha$ . We first consider the limit of  $q_p \ll 1$  and  $\omega q_p \lesssim 1$ . In this limit  $C(c_1)$  is small as long as  $c_1 < k$ . The numerator of 5 can then be approximated by  $-B(c_1) + |B(c_1)|$ . As a result we can distinguish two regimes depending on the sign of  $B(c_1)$ :

$$\frac{c_3^*(c_1)}{k^*} \approx \begin{cases} 0 & \text{if } c_1 < c_1^b, \\ \frac{|B(c_1)|}{A(c_1)} & \text{if } c_1 > c_1^b, \end{cases} \quad (6)$$

where  $c_1^b$  is the border between the two regimes, implicitly given by  $B(c_1^b) = 0$ :

$$c_1^b = \frac{k}{\omega} \frac{r(q_p + 1) - q_p}{q_p(\omega - r) - rk/\omega}. \quad (7)$$

Note that  $c_1^b \ll k$  provided  $r \ll \omega$ ; under this condition approximation 6 holds around the transition  $c_1 = c_1^b$ . As both  $B(c_1)$  and  $A(c_1)$  are linear functions of  $c_1$ , the second regime has the form of a Hill function with Hill coefficient  $n=1$ . Therefore equation 6 shows that equation 5 behaves like a sharp threshold response. This threshold effect is responsible for the increased steepness of the response due to conditional auto-activation.

The maximal expression following from equation 5, at full activation, is  $\omega q_p / (1 + \omega q_p)$ . This demonstrates that in the limit of  $q_p \ll 1$  the maximum expression level becomes very low (for a given value of  $\omega$ ). On the other hand, if  $q_p$  is increased the term  $4A(c_1)C(c_1)$  becomes more and more significant and the transition between the two regimes in equation 6 becomes more and more gradual. Consequently, in the optimized functions plotted in Fig. 2 the values of  $q_p$  reflect a compromise between the opposing requirements of having a high maximal expression (requiring a large  $q_p$ ) and a sharp threshold response (requiring a small  $q_p$ ).

The steepness of a function  $f(x)$  in the point  $x$  can be formalized by the *sensitivity*, defined by  $S(x) \equiv |d \log f(x) / d \log x| = |x f'(x) / f(x)|$ . The sensitivity of a Hill function is limited by the Hill coefficient  $n$ , which is equal to the number of cooperatively interacting binding sites for the input TF. We therefore ask if a similar limitation applies to the minimal model of conditional auto-activation. From equation 5 the sensitivity function  $S(c_1)$  can be derived straightforwardly. The result is rather cumbersome and therefore an exact expression for the *maximal* sensitivity is hard to obtain. However, since the most sensitive part of the function is in the region where  $B(c_1) \approx 0$  (i.e., close to  $c_1^b$ ), the maximal sensitivity can be approximated by  $S(c_1^b)$ . In the limit of large  $\omega$  this expression converges to

$$\lim_{\omega \rightarrow \infty} S(c_1^b) = \frac{(1 - q_p(r-1))(\sqrt{q_p(q_p+r+1)} - q_p)}{2q_p(q_p+r+1)}. \quad (8)$$

Importantly, since we evaluated the sensitivity in a point close to but not exactly at the maximum, this approximate result is a conservative estimate: the true maximum cannot be lower than this. In the limit of small  $q_p$  the maximum sensitivity diverges as  $1/\sqrt{4q_p(1+r)}$ , which proves that the sensitivity of response function 5 does not have a theoretical upper limit, unlike those of Hill functions.

So far we have neglected the dynamical properties of the designs because the current model only considers the steady-state response of the system. As we mentioned, auto-activation tends to slow down the response time of the system. Therefore, in systems where the speed of response is of great importance cooperative regulation is expected to outperform conditional auto-activation. Selection on response speed is discussed in more detail below.

### Auto-activation can sharpen repression

A second feedback pattern emerges in logic gates in which repression is important, notably the NAND, NOR and IN (inhibition) gates. As it turns out, whenever steep repression is required, we also find strong auto-activation; this occurred in every simulation run for our NAND, NOR and IN gates (20 repeats each), with  $M_{AR} > 6.9 \times 10^4 \text{ nM}^2$  in all cases. We present the NAND gate as an example.

Fig. 3A shows the *cis*-regulatory region of a typical NAND gate using auto-activation. The corresponding response function plotted in Fig. 3B indeed shows an excellent NAND-like behavior. As quantified below, in fact it performs better than the design without auto-activation reported earlier and reproduced for comparison in Fig. 3A and B [15].

The design that resulted when auto-activation was excluded is composed of a hetero-cooperative repression module (a tandem series of repressor sites to which both input TFs bind cooperatively). The function of this module is to repress transcription only when both TF1 and TF2 are present in sufficiently high concentrations, as required of a NAND gate. In Ref. [15] we pointed out that in the simulation results such a repression module was always accompanied by strong *activation* sites for both input TFs. This counter-intuitive feature turned out to enhance the sharpness of the response. At low TF concentrations, the activation sites counter-act the repression module, so that the expression stays high. At higher TF concentrations, however, the repression module dominates and represses transcription. Under the parameters used designs of this type reached a modest fold-change of  $\approx 6$  and a deviation measure  $M_{RF} = 8.5 \times 10^5 \text{ nM}^2$  (see section ‘‘Evolutionary algorithm’’ in the methods section for the definition of  $M_{RF}$ ).

In the new results (Fig. 3) the activation sites for TF1 and TF2 have disappeared, but instead we find auto-activation. In the absence of TF1 and TF2,  $tf3$  is highly expressed, aided by auto-activation. As the concentrations of TF1 and TF2 are increased, the repression module starts to compete with the auto-activation module. Quite suddenly, the repression module wins this competition and displaces RNAP from the promoter. The strong, cooperative repression module now leads to a rather complete inhibition. The new design can lead to fold-changes of  $\approx 30$  and a deviation measure of  $M_{RF} = 1.2 \times 10^5 \text{ nM}^2$ .

To study the mechanism responsible for the steepness of the response function we again analyzed a minimal model. In Fig. 3C and D two scenarios for an IN gate are compared. In the first scenario, a transcription factor TF1 cooperatively binds to a pair of repressor sites to inhibit the gene  $tf3$ . In the second scenario we use the same configuration, but add an activator site for TF3. Thus, auto-activation competes with cooperative repression. The fitness of each design is optimized using the fitness function for the IN gate. As can clearly be seen in Fig. 3D the second scenario, using conditional auto-activation, results in a steeper and more complete repression. Figure 1 in Text S1 shows plots of the sensitivity  $S(c_1)$  as a function of  $c_1$  for the response plots in Fig. 3D and clearly demonstrates that auto-activation enhances the sensitivity.

Does the sensitivity of the response function 9 have an upper bound, as is the case for Hill functions? To answer this question we again study the minimal model mathematically.

The response function  $c_3^*(c_1)$  of the minimal model is given by

$$\frac{c_3^*(c_1)}{k^*} = \frac{-B(c_1) + \sqrt{B(c_1)^2 + 4A(c_1)C(c_1)}}{2A(c_1)}, \quad (9)$$

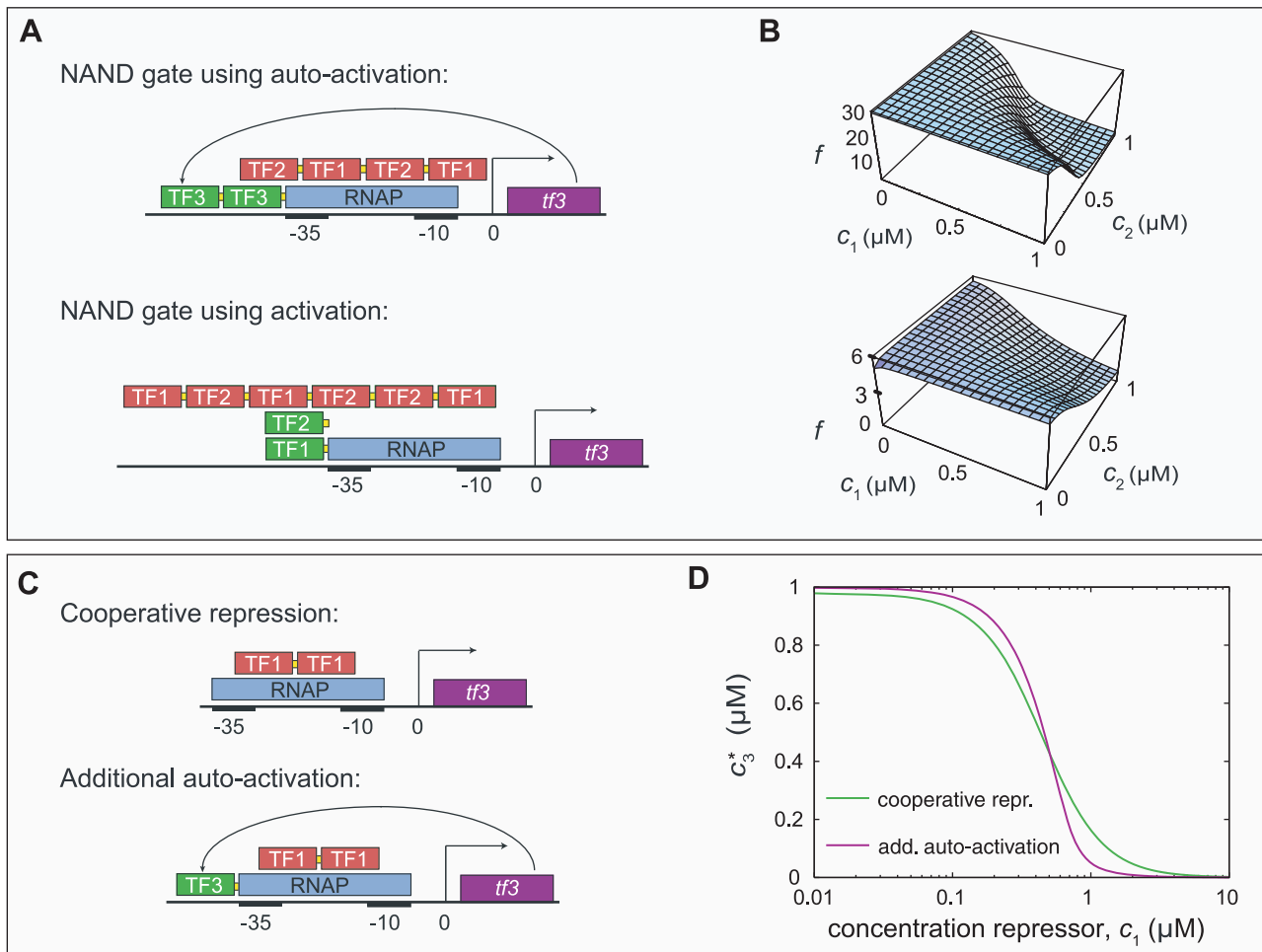
with

$$A(c_1) \equiv 1 + \omega q_p + 2c_1/k + \omega(c_1/k)^2, \quad (10)$$

$$B(c_1) \equiv 1 - q_p(\omega r - 1) + 2c_1/k + \omega(c_1/k)^2, \quad (11)$$

$$C(c_1) \equiv q_p r. \quad (12)$$

We first describe the limit in which  $q_p, \omega \gg 1$  and  $r \approx 1$ . The form of this equation is obviously similar to equation 5 for conditional auto-activation. However, here  $B(c_1)$  is large and *negative* ( $\approx -q_p \omega$ ) when



**Figure 3. Sharp and complete repression using auto-activation.** In the simulations, auto-activation evolved in every gate that requires strong repression. This figure shows two examples in which auto-activation indeed aids sharp and thorough repression. Fig. A and B depict NAND gates resulting from the simulations. When auto-regulation is not allowed by the method, the input TFs have both activating and repressing binding sites, as reported earlier [15] (in Fig. A and C, red boxes represent repressor sites and green boxes activator sites). When auto-regulation is included (*i.e.*, the regulated gene *tf3* codes for a transcription factor TF3 that can bind to its own *cis*-regulatory region) auto-activation emerges. Gene *tf3* is still repressed by a hetero-cooperative module consisting of binding sites for TF1 and TF2. At low concentrations of TF1 and TF2 the auto-activation counteracts the repression module; as a consequence, the response to the concentrations of TF1 and TF2 is very sharp and the fold-change  $f$  high, as can be seen in Fig. B. In Fig. C and D we study the mechanism in a simpler model system. Fig. C shows the *cis*-regulatory regions of two slightly different repression systems. In both cases, a transcription factor TF1 represses a gene *tf3*, coding for a second transcription factor TF3, by binding cooperatively to a pair of repressor sites. In the second scenario, an auto-activation site for TF3 is present as well. Fig. D presents the steady-state expression level of *tf3* as a function of the repressor concentration. In both alternatives we optimized the binding sites using the fitness function described in the main text. Clearly, the second scenario leads to a more sensitive repression curve than the first. The presence of auto-activation allows for stronger repressor sites; consequently, as the concentration of TF1 increases the displacement of RNAP from the promoter by the repressor is more effective (*i.e.*, the remaining expression level at  $c_1^* > 0.2$  is much lower than in the cooperative repression case). doi:10.1371/journal.pcbi.1000813.g003

$c_1 \lesssim k$ . In this regime,  $B(c_1)^2 \approx (q_p \omega)^2 \gg 4A(c_1)C(c_1)$ , so that the numerator is approximated by  $2|B(c_1)|$ . As  $c_1$  increases,  $|B(c_1)|$  decreases while the denominator increases; therefore the expression is rapidly repressed. This regime ends suddenly as  $B(c_1)$  reaches zero, at  $c_1 = c_1^b$ ; at this point the expression is almost fully repressed.

The sensitivity function  $S(c_1)$  can be derived from equation 9. The exact expression is again too cumbersome to derive the maximal sensitivity analytically. However, the most sensitive region of the response plot is again expected around  $B(c_1) \approx 0$  so that we estimate the maximal sensitivity as

$$S(c_1^b) = \frac{(\sqrt{r(1+r)\omega-r}) + r}{q_p r \omega (\omega(r+1)-1)} (q_p r \omega^2 - \sqrt{q_p r \omega^2 - (q_p+1)\omega+1} - (q_p+1)\omega+1) \quad (13)$$

For large  $q_p$  this converges to

$$\lim_{q_p \rightarrow \infty} S(c_1^b) = \frac{(r\omega-1)(\sqrt{r(r+1)\omega-1}+r)}{r(r+1)\omega-1}, \quad (14)$$

which for large  $\omega$  approaches  $1 + \sqrt{\omega}$ . Numerical tests demonstrate that this conservative approximation becomes excellent for  $\omega \gg 1$  (data not shown). In the absence of auto-regulation, the sensitivity cannot exceed 2, the number of repressor sites. Equation 14 demonstrates that in the presence of auto-regulation the sensitivity can easily exceed 2 but is nevertheless limited given  $\omega$ .

We note that the sensitivity is optimal for  $q_p \gg 1$ . Hence, unlike the case of conditional auto-activation the requirements of a high



maximal expression and a high sensitivity do not contradict. In the simulations as well as in reality, however, the promoter strength is bounded by other factors. Clearly the binding affinity of RNAP for the promoter is bounded by the physics of RNAP–DNA binding. A less obvious constraint follows from the fact that the expression switches from high to low around  $c_1^b = k\sqrt{1-\omega+\omega q_p(\omega r-1)}/\omega$ ; if the repression is to occur at reasonable TF1 concentrations (the simulations impose the interval [333nM,667nM]) high values of  $q_p$  require low values of the dissociation constant  $k$  (*i.e.*, strong repression). Finally, a high sensitivity in one point does not guarantee that the response function switches from high to low in a narrow interval as is required by the fitness function; this explains why in the plots in Fig. 3 the maximal sensitivity is not optimal ( $S_{\max} = 3.7 < 1 + \sqrt{w} = 6.5$ ).

### Linear repression benefits from auto-activation

In both previous cases, auto-regulation was used to obtain the steep or switch-like behavior required to approximate the binary responses of logic gates. Indeed, sharp responses are observed and probably required in many real examples; nevertheless many genes respond in a more gradual manner to their input signals (Ref. [17] provides examples of both sharp and gradual responses). Is auto-regulation also useful in cases where a gradual response is required? To test this, we now turn to the results of simulations with *linear* goal functions.

Indeed, simulation results for linear repression (*i.e.*, the LIN and NMEAN gates) always use auto-regulation, with  $M_{AR}$  in the range  $1.2 \times 10^4 - 1.6 \times 10^5 \text{ nM}^2$ . As can be seen in Fig. 4, approximately linear repression can be obtained when repression is combined with auto-activation; the deviation measure for the simulation result shown is  $M_{RF} = 3.7 \times 10^3$ . The same figure shows results of simulations in which auto-regulation is excluded. In that case a large cooperative repression module results, which leads to a less linear result ( $M_{RF} = 2.1 \times 10^4$ ).

Again, we analyzed the mechanism through a slightly simplified model presented in the same figure. The promoter design of the simplified model is identical to the one presented in Fig. 3C, where it was used to demonstrate how auto-activation can provide sharp

responses. In essence, the difference between the two cases is that in the IN gate the two repressor sites have the same affinity, whereas in the LIN case they do not: one of them is many times weaker than the other ( $k_D = 38 \text{ nM}$  vs.  $k_D = 7.17 \mu\text{M}$ ). As a result, the repression is introduced gradually as the repressor concentration increases.

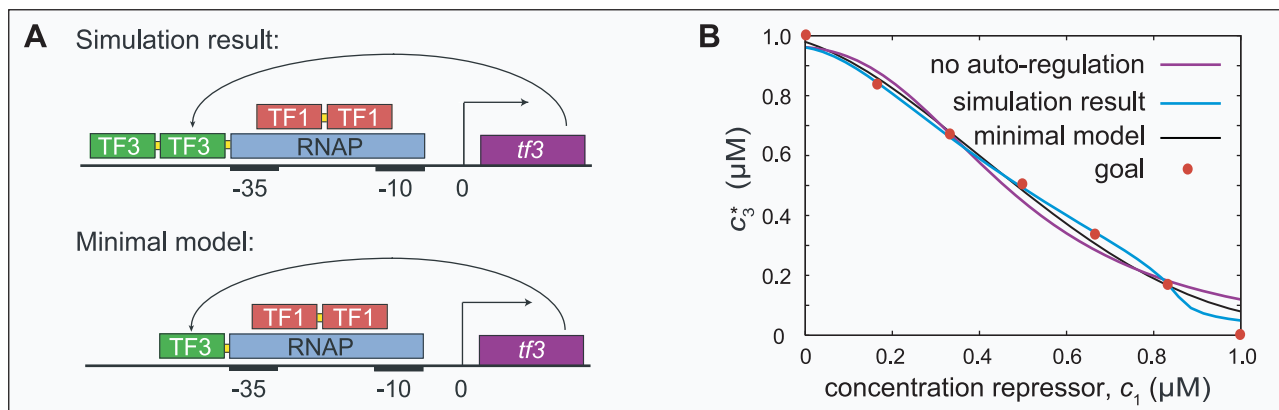
Linear repression requires that  $dc_3^*/dc_1 = -1$  in the domain  $[0, 1000 \text{ nM}]$ . Since  $c_3^*$  is defined as the solution of  $\alpha p_{\text{on}}(c_1, c_3^*) = \beta c_3^*(c_1)$ , we can take the total derivative of this relation to arrive at

$$\frac{dc_3^*}{dc_1} = \frac{\alpha \partial p_{\text{on}}(c_1, c_3^*)}{\beta \partial c_1} / \left( 1 - \frac{\alpha \partial p_{\text{on}}(c_1, c_3)}{\beta \partial c_3} \Big|_{c_3^*} \right). \quad (15)$$

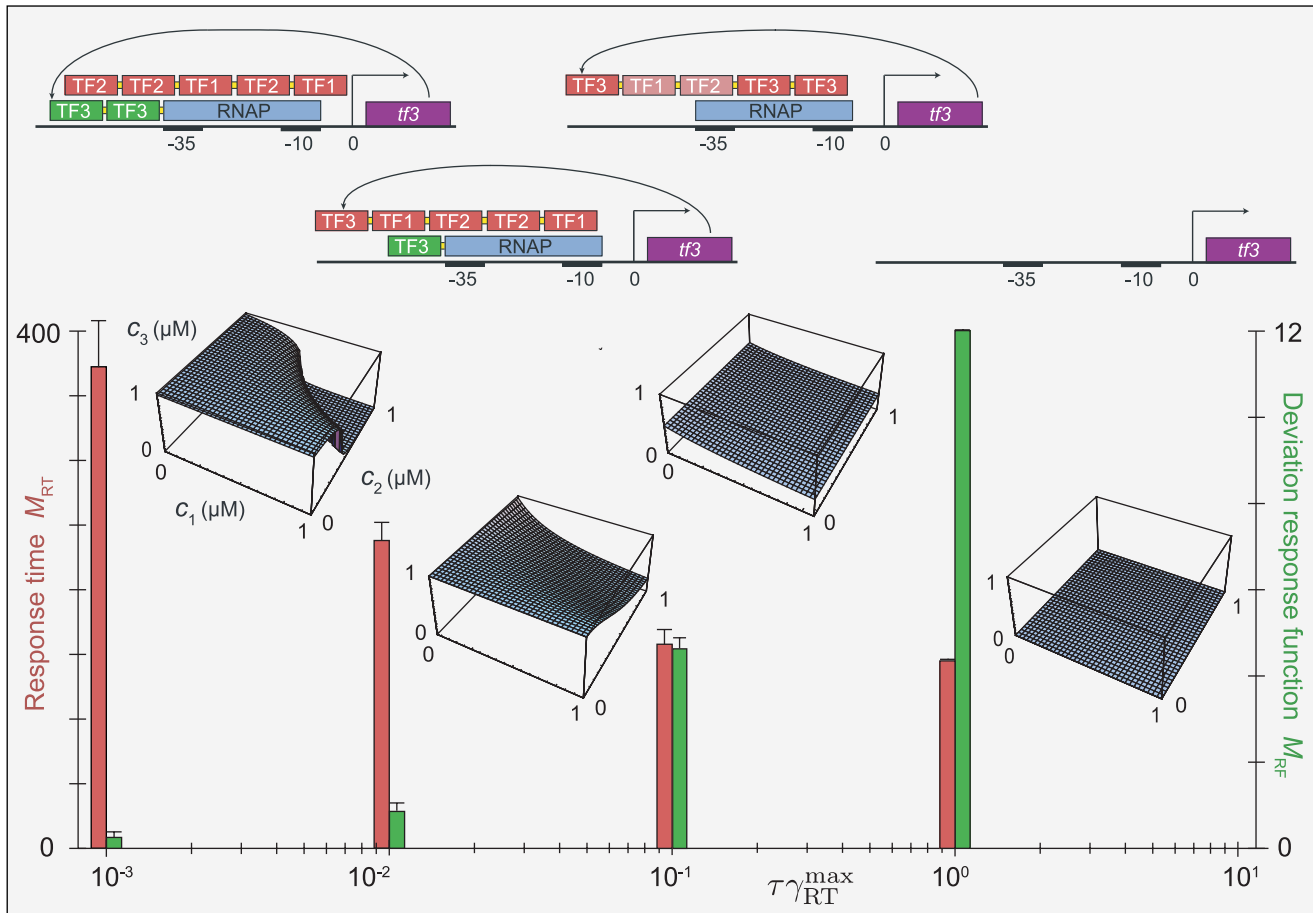
In the absence of auto-regulation the denominator equals 1. In this case  $p_{\text{on}}$  is a Hill-type function of  $c_1$  and therefore its derivative is not constant. In the presence of auto-regulation the denominator can be used to correct some of the variation in the numerator. (See Text S1.)

In contrast, in the simulation results for linear *activation* (both the LACT and the MEAN gates) auto-regulation is never used. To test if these results are an artifact of the algorithm, we studied a series of models (see Text S1). Each model is a possible layout of transcription factor binding sites and includes auto-regulatory sites. For each of the models, we optimized the affinities of all binding sites with respect to the fitness score, using a standard Nelder–Mead optimization routine. Consistent with our simulations, in the solutions for all models the affinities of the auto-regulatory sites vanished. Even though the list of models tested is not exhaustive, this suggests that auto-regulation is not helpful in constructing LACT or MEAN gates.

To illustrate an important difference between linear activation and linear repression we provide the following general argument. Suppose that an accurate LACT gate can be constructed using auto-regulation. By definition the response function should then be  $c_3^* = c_1$  in the interval  $[0, 1000 \text{ nM}]$ . Consequently,  $c_3^*(c_1) = (\alpha/\beta)p_{\text{on}}(c_1, c_3^*) = (\alpha/\beta)p_{\text{on}}(c_1, c_1)$  in this interval. Interestingly, this shows that if all TF3 binding sites in the *cis*-regulatory



**Figure 4. Linear repression using auto-regulation.** If in the simulations we selected for a linearly decreasing response function (a LIN gate), auto-activation emerged. The resulting *cis*-regulatory region is schematically depicted in Fig. A. Red boxes and green boxes represent repressor and activator sites, respectively. The corresponding response function is plotted in Fig. B, alongside the results of a simulation in which auto-regulation is excluded. The auto-activation indeed manages to straighten the repression curve. The seven red dots in Fig. B show the goal function that is used in the fitness function: gates are considered better if their response function fits these points better (see Methods). We again studied a simplified model in more detail; the *cis*-regulatory region and response function of this minimal model are also shown. Indeed, the simple model system with appropriate binding site affinities fits the goal points better than the design without auto-regulation.  
doi:10.1371/journal.pcbi.1000813.g004



**Figure 5. NAND gates at increasing selection pressure on response speed.** On the one hand, the sensitivity of the response function of a NAND gate is improved by auto-activation. On the other hand, the response speed of the gate is enhanced by auto-repression. Consequently, if selection acts both on the response function and the response time, the simulation results are a compromise and depend critically on the relative magnitudes of the two selection pressures. The figure shows representative response functions and promoter designs of NAND gates resulting from four values of the parameter  $\gamma_{RT}^{\max}$ , which controls the weight of the response speed in the total fitness function  $F_{\text{tot}} = C - \gamma_{RF} M_{RF} - \gamma_{RT}(t) M_{RT}$ . (The irrelevant constant  $C$  merely serves to ensure that  $F_{\text{tot}} > 0$ .) The average values of the measures  $M_{RT}$  (measuring the response time in arbitrary units  $\tau$ ) and  $M_{RF}$  (the deviation of the response function from the goal function in units  $\mu\text{M}^2$ ) for each condition are also plotted. By definition, low values of  $M_{RT}$  and  $M_{RF}$  correspond to *good* performance. For the lowest value  $\gamma_{RT}^{\max} = 10^{-3}/\tau$  the response function is optimized and shows an excellent NAND gate. Due to strong, cooperative auto-regulation, the response is very sharp and almost bistable in the transition region, but the response speed is low. At  $\gamma_{RT}^{\max} = 10^{-2}/\tau$  the result is a compromise: the quality of the response function is clearly reduced but the response speed is higher. Still auto-activation evolves, but it is weaker and non-cooperative and combined with weak auto-repressing binding sites. At  $\gamma_{RT}^{\max} = 10^{-1}/\tau$  auto-repression fully takes over; the response function is crippled but the response speed is high. If the selection pressure on response time is increased even further ( $\gamma_{RT}^{\max} = 10^0/\tau$ ) the response speed is fully optimized by disabling the response altogether.  
doi:10.1371/journal.pcbi.1000813.g005

region are replaced by binding sites for TF1—resulting in a gate without auto-regulation—the *exact same response function should be obtained*. Even though this argument does not prove that auto-regulation cannot be used to construct LACT gates, it does show that if a high-quality LACT can be constructed with auto-regulation a similar response can always be obtained without it as well. This is in stark contrast with the linear repression case, where  $c_3^* \neq c_1$ .

### Selection for quick responses or against noise leads to auto-repression

Surprisingly, auto-repression does not show up in any of the simulations described so far, whereas auto-activation features regularly. As we mentioned in the introduction, previous studies have shown that negative auto-regulation can be used to diminish intrinsic noise and to speed up response times. In the simulations presented so far, such qualities were not rewarded. Therefore we

asked if auto-repression would emerge if we did select for such dynamic properties on top of our usual selection criteria.

First, we used a heuristic measure  $M_{RT}$  (where RT stands for Response Time) to select for a quick response to changes in the input parameters; it was computed as follows. For 16 combinations of input concentrations ( $c_1, c_2$ ) (corresponding to the red dots in Fig. 2) we numerically solved the differential equation 4 with two different initial conditions:  $c_3 = 0\mu\text{M}$  and  $c_3 = 1\mu\text{M}$ . The solutions were used to measure the time it took for the system to approach the steady-state value up to a small distance  $\epsilon = 10^{-4}\mu\text{M}$ . The measure  $M_{RT}$  was defined as the sum of all 32 response times. The total fitness function, combining selection on the response function with selection on the response time, was  $F_{\text{tot}} \equiv C - \gamma_{RF} M_{RF} - \gamma_{RT} M_{RT}$ , where the factor  $\gamma_{RT}$  was used to tune the relative strength of the selection on the response time. Again,  $C$  is an irrelevant constant ensuring that  $F_{\text{tot}} > 0$ .

In fact, the initial condition of the simulations, in which the gate is completely dysfunctional, is a local optimum of this fitness score. This is because initially the steady-state expression level is negligible so that the response time for initial condition  $c_3 = 0\mu\text{M}$  is practically zero. Even though mutations that increase the constitutive promoter activity improve both the response function and the response time for initial condition  $c_3 = 1\mu\text{M}$ , the concomitant increase in the response time for initial condition  $c_3 = 0\mu\text{M}$  dominates. To ensure that the simulation was not trapped in this local optimum  $\gamma_{\text{RT}}$  was increased slowly from 0 to  $\gamma_{\text{RT}}^{\text{max}}$  in the course of the simulations, according to:

$$\gamma_{\text{RT}}(t) = \gamma_{\text{RT}}^{\text{max}} \frac{t^3}{t_{\text{in}}^3 + t^3}, \quad (16)$$

where  $t$  is the simulation time (*i.e.*, the cycle number of the evolutionary algorithm), and  $t_{\text{in}} = 50$ .

Second, we selected against intrinsic noise, *i.e.*, fluctuations in the concentration of  $c_3$  due to the stochasticity of the processes involved in the production and degradation of TF3. This type of noise should be contrasted with extrinsic noise, which here is understood to be the noise due to fluctuations in the input concentrations  $c_1$  and  $c_2$  or due to changes in RNAP concentrations [12,27]. Even though extrinsic noise is generally important too [28], the treatment of extrinsic fluctuations involves subtleties that are beyond the scope of this work, such as the question which changes in  $c_1$  and  $c_2$  should be considered changes in the input signal and which should be considered noise. We therefore only consider intrinsic noise.

In order to treat intrinsic fluctuations in a tractable manner we now replaced the ordinary differential equation 4 by the following stochastic differential equation:

$$\frac{dc_3(t)}{dt} = \alpha p_{\text{on}}(c_1, c_2, c_3(t)) - \beta c_3(t) + \xi(t). \quad (17)$$

The term  $\xi(t)$  represents Gaussian white noise and is characterized by (see Text S1):

$$\langle \xi(t') \xi(t) \rangle = (\alpha p_{\text{on}}(c_1, c_2, c_3(t)) / V + \beta c_3(t) / V) \delta(t' - t). \quad (18)$$

The first term on the right-hand side describes the noise in the production of TF3 while the second term describes the stochasticity in the degradation of TF3. Both terms depend explicitly on the volume  $V$  because, at constant concentration, the copy number of TF3 scales with  $V$  which affects the variance in  $c_3$ .

In Text S1 we show that the standard deviation  $\sigma$  of the concentration  $c_3$  can be approximated as:

$$\sigma(c_3^*)^2 \approx \frac{1}{b(c_3^*)}, \quad (19)$$

with

$$b(c_3^*) \equiv V \frac{1 - (\alpha/\beta)p'_{\text{on}}(c_1, c_2, c_3^*)}{c_3^*} - \left( \frac{(\alpha/\beta)p'_{\text{on}}(c_1, c_2, c_3^*) + 1}{2c_3^*} \right)^2 + \frac{(\alpha/\beta)p''_{\text{on}}(c_1, c_2, c_3^*)}{2c_3^*}, \quad (20)$$

where  $p'_{\text{on}}$  and  $p''_{\text{on}}$  are the first and second partial derivatives of  $p_{\text{on}}$  with respect to  $c_3$ , and  $c_3^*$  is the steady-state solution of the

deterministic equation 4. We computed the right-hand side of equation 19 numerically for 16 input values ( $c_1, c_2$ ) (again, corresponding to the red dots in Fig. 2) and treated the sum of the results as an additional fitness measure  $M_{\text{N}}$  (where N stands for Noise). The strength of selection against noise was again increased gradually during the simulations (analogous to equation 16). The total fitness function thus became  $F_{\text{tot}} \equiv C - \gamma_{\text{RF}} M_{\text{RF}} - \gamma_{\text{N}}(t) M_{\text{N}}$ .

We performed simulations with several values for  $\gamma_{\text{RT}}^{\text{max}}$  and  $\gamma_{\text{N}}^{\text{max}}$ :  $\tau\gamma_{\text{RT}}^{\text{max}} \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^0\}$  (where  $\tau$  is the arbitrary unit of time, see Methods), and  $\gamma_{\text{N}}^{\text{max}} \in \{10\mu\text{M}^{-1}, 20\mu\text{M}^{-1}, 40\mu\text{M}^{-1}, 80\mu\text{M}^{-1}\}$ . Indeed, in these simulations auto-repression emerged.

In activating gates (ACT, AND, OR) auto-repression resulted in all simulation runs with  $\gamma_{\text{N}}^{\text{max}} = 10\mu\text{M}^{-1}$  or  $\tau\gamma_{\text{RT}}^{\text{max}} = 0.01$ . The auto-repression was invariably strong, with  $M_{\text{AR}} > 5.6 \times 10^4$ , and mediated by multiple cooperative binding sites. If  $\gamma_{\text{N}}^{\text{max}}$  or  $\gamma_{\text{RT}}^{\text{max}}$  were further increased eventually the resulting *cis*-regulatory regions became completely dysfunctional; this can be understood from the fact that both the response time and the noise reduction can be optimized by abolishing expression altogether. Figure 7 in Text S1 demonstrates how the properties of resulting OR gates changed as a function of  $\gamma_{\text{N}}^{\text{max}}$ . As  $\gamma_{\text{N}}^{\text{max}}$  is increased the deviation of the response from the ideal OR gate, measured by  $M_{\text{RF}}$ , increases, while the noise, measured by  $M_{\text{N}}$ , decreases.

As we explained, the response functions of the NAND, NOR and IN gates benefit from auto-activation; in those gates auto-activation occurred unless the selection pressure on the dynamical properties dominated (*i.e.*, if  $\gamma_{\text{N}}^{\text{max}}$  or  $\gamma_{\text{RT}}^{\text{max}}$  were large), in which case the quality of the response functions was negatively affected. Fig. 4 shows results from simulations selecting for NAND gates at various values of  $\gamma_{\text{RT}}^{\text{max}}$ . As the selection pressure on response time was increased the response functions became more and more compromised. Auto-activation resulted for  $\tau\gamma_{\text{RT}}^{\text{max}} = 10^{-3}$  and  $\tau\gamma_{\text{RT}}^{\text{max}} = 10^{-2}$ ; in the former case the promoter designs were of the type shown in Fig. 3A, while in the latter case only one auto-activation site remained. Interestingly, in most of the simulation runs (18 out of 20) at  $\tau\gamma_{\text{RT}}^{\text{max}} = 10^{-2}$  a weak auto-repression site shows up in conjunction with the auto-activation site. These weak auto-repression sites are incorporated in the hetero-cooperative repression module and have a high occupancy only at high concentrations of TF1 and TF2; analogous to conditional auto-activation, this effect could be called conditional auto-repression. At  $\tau\gamma_{\text{RT}}^{\text{max}} = 10^{-1}$  the auto-activation was replaced by strong auto-repression mediated by a single or multiple binding sites and the sites for the input TFs were very weak. Finally, at  $\tau\gamma_{\text{RT}}^{\text{max}} = 1$  the resulting gates became completely dysfunctional and no significant binding sites remained.

### Auto-activating TFs have more inputs

The results above suggest that auto-activation and auto-repression have very different functions. We therefore wondered whether in the known transcription regulatory network of *E. coli* the auto-activators and auto-repressors have different statistical properties.

Surprisingly, we found that auto-activators are more often regulated by other TFs than auto-repressors. According to the data in RegulonDB [2], 18 of the 25 auto-activating TFs in *E. coli* are regulated by at least one additional TF (72%) versus 30 out of 62 auto-repressing TFs (48%); this indeed suggests that auto-activators are more likely to have additional inputs ( $p = 0.037$ ). The difference becomes more convincing if we look at the total number of inputs for the two sets. The 25 auto-activators have, in total, 52 inputs (*i.e.*, an in-degree of 2.08 on average; the auto-regulation is not counted as an input) while the 62 auto-repressors have 50 inputs in total (0.81 on average). Evidently, auto-activators have significantly more inputs than auto-repressors ( $p = 0.0013$ ).

Since auto-regulation can potentially have many functions and most of the auto-regulators are poorly characterized, we can only speculate about the origin of this difference. One possible explanation would be the following. If a common function of auto-activation is to shape response functions, as suggested by our analysis, then auto-activation should evolve preferentially for TFs that are regulated by one or more input TFs. In that case one would expect the average in-degree for auto-activators to be high. The same argument does not hold for auto-repression: our results suggest that auto-repression typically evolves for different reasons. Some of the functions of auto-repression suggested in the literature, such as its tendency to decrease intrinsic noise and to mitigate the effect of changes in the bacterial growth rates on gene expression, do not require additional input TFs. It is therefore not too surprising that for many auto-repressors (32 out of the 62) no input TF is known.

## Discussion

Our results shed new light on the use of auto-regulation. We described three situations in which auto-activation can be used to improve the response function of promoters. First, if auto-activation is conditional on the presence of other TFs, it can give rise to sensitive responses that otherwise require multiple cooperative binding sites of the input TF. Presumably, not all input TFs can bind cooperatively to multiple binding sites; in those cases conditional auto-activation can serve as an alternative. Secondly, auto-activation can strongly contribute to the sensitivity of the response of repression systems. Whenever sharp repression is required, auto-activation can have a selective advantage. Thirdly, we showed that auto-activation is also useful if a linearly decreasing response function is desired. Together, such mechanisms may help explain the large number of auto-activators present in *E. coli*.

Auto-repression never appeared in the simulation results if selection was based on the response function of the gates only. Most likely, the limited use of auto-repression in shaping response functions is due to its general tendency to decrease the fold-change and sensitivity of the response. A low fold-change or sensitivity can typically also be achieved without auto-repression by tuning both the promoter strength and the affinities of the TF binding sites. Nevertheless, we cannot exclude the possibility that auto-repression would show up in simulations selecting for response functions different from the ones presented here.

If the fitness function was altered to favor a fast response or suppression of intrinsic transcriptional noise, auto-repression did emerge. It has been suggested before that the function of negative auto-regulation is to regulate such dynamic properties [5,9,10]; our results support this view.

In retrospect, the emergence of auto-regulation is hardly surprising. The evolution of *cis*-regulatory regions can be perceived as adaptive curve fitting. Allowing for auto-regulation gives gene-regulatory systems additional degrees of freedom to optimize their performance, and it would perhaps be more surprising if this freedom were not exploited. We therefore expect that the conclusions based on the idealized gates studied in this work are also relevant for real biological systems requiring more complex response functions.

We have seen that in some cases the advantage of using auto-regulation is large (*e.g.* when sensitive repression is required) whereas in other cases there is only a small difference between the quality of the response function for designs with or without auto-regulation. This leads one to wonder whether in the latter case natural selection on the shape of the response would be large enough to evolve and maintain auto-regulation, in particular in the presence of noise. This is largely an open question; yet, the fact that some *E. coli* promoters

contain a large number of TF binding sites many of which contribute only marginally to the expression (see for instance [29]) suggests that, at least in some *cis*-regulatory regions, natural selection is strong enough to fine-tune the response function in great detail.

The results presented are quite insensitive to the parameter values chosen. The value of  $\omega$  influences important properties such as the maximum fold change in activation systems, but as long as it is chosen within the biological range 10–100 the designs of the gates do not seem to depend qualitatively on the value chosen. To verify this, we performed simulations with  $\omega = 100$  for AND, NAND, NOR and OR gates (without selection against noise or response speed) and found the results to be qualitatively the same as those presented. The value of  $d$  influences the spacing of binding sites within a module, but not the basic design properties, as long as  $d \lesssim M/2$  so that overlapping modules can be constructed that bind independently. The results are also insensitive to the length of the binding sites  $M$  (we tested this with simulations for AND, NAND, NOR and OR gates with  $M = 15$ ) and the matrix elements of the binding energy matrix; essential is only that the evolutionary algorithm can tune the dissociation constants of the binding sites to a wide range of values (1–10000nM), as in reality. The length of the *cis*-regulatory region,  $N$ , determines the maximum number of tandem binding sites that fit on the regulatory region; larger values of  $N$  therefore ultimately lead to larger tandem arrays. However, since tandem arrays of five or more binding sites can form in the simulations, we believe that  $N = 100$  is large enough to accommodate typical *E. coli* promoters.

Even though in eukaryotes the mechanisms of gene regulation are generally different and various additional layers of regulation exist, recent work has shown that many basic principles of prokaryotic gene regulation—in particular the interplay between cooperative binding and competitive inhibition—are equally important in eukaryotes (see for instance [30] about repression and inhibition in yeast and [31] about enhancers in *Drosophila*). Auto-regulation is also widespread in eukaryotes [32]; therefore, our findings could also be relevant for gene regulation in eukaryotes.

As we mentioned, auto-activation is known to reduce the response speed in some situations and to increase the amplitude of fluctuations. Clearly, those issues may be problematic in some real-life situations. On the other hand, a slow response can be a positive feature as well if it is applied as a filter of high-frequency noise (a low-pass filter). Fluctuations may in some cases be beneficial or even necessary. For instance, when cells respond to a fluctuating environment via the strategy of stochastic switching, fluctuations are essential [33]. But even when cells cope with a fluctuating environment via the strategy of deterministic switching, fluctuations may be beneficial, since they can increase the population's growth rate when the response function is suboptimal [34]. Indeed, the fact that auto-activation is found so often in *E. coli* demonstrates that the associated reduction of the response speed and the amplification of fluctuations can apparently be circumvented, tolerated or put to use.

## Supporting Information

**Text S1** Supplementary information. All supplementary information is contained in this document. It contains supplementary figures, detailed mathematical derivations, additional analyses, and numerical tests of approximations used in the main text.

Found at: doi:10.1371/journal.pcbi.1000813.s001 (0.81 MB PDF)

## Author Contributions

Conceived and designed the experiments: RH BU PRtW. Performed the experiments: RH BU. Analyzed the data: RH BU PRtW. Wrote the paper: RH.

## References

1. Keseler IM, Collado-Vides J, Gama-Castro S, Ingraham J, Paley S, et al. (2005) EcoCyc: a comprehensive database resource for *Escherichia coli*. *Nucl Acids Res* 33: D334–337.
2. Salgado H, Santos-Zavaleta A, Gama-Castro S, Millan-Zarate D, Diaz-Peredo E, et al. (2001) RegulonDB (version 3.2): transcriptional regulation and operon organization in *Escherichia coli* k-12. *Nucl Acids Res* 29: 72–74.
3. Thieffry D, Huerta AM, Perez-Rueda E, Collado-Vides J (1998) From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in *Escherichia coli*. *Bioessays* 20: 433–440.
4. Shen-Orr SS, Milo R, Mangan S, Alon U (2002) Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nat Genet* 31: 64–68.
5. Becskei A, Serrano L (2000) Engineering stability in gene networks by autoregulation. *Nature* 405: 590–593.
6. Thattai M, van Oudenaarden A (2001) Intrinsic noise in gene regulatory networks. *Proc Natl Acad Sci U S A* 98: 8614–9.
7. Stekel DJ, Jenkins DJ (2008) Strong negative self regulation of prokaryotic transcription factors increases the intrinsic noise of protein expression. *BMC Syst Biol* 2: 6.
8. Klumpp S, Zhang Z, Hwa T (2009) Growth rate dependent global effects on gene expression in bacteria (in press). *Cell*.
9. Savageau MA (1974) Comparison of classical and autogenous systems of regulation in inducible operons. *Nature* 252: 546–549.
10. Rosenfeld N, Elowitz MB, Alon U (2002) Negative autoregulation speeds the response times of transcription networks. *J Mol Biol* 323: 785–793.
11. Elowitz MB, Leibler S (2000) A synthetic oscillatory network of transcriptional regulators. *Nature* 403: 335–338.
12. Hornung G, Barkai N (2008) Noise propagation and signaling sensitivity in biological networks: A role for positive feedback. *PLoS Computational Biology* 4: e8.
13. Nevzhay D, Adams RM, Murphy KF, Josic K, Balázsi G (2009) Negative autoregulation linearizes the dose-response and suppresses the heterogeneity of gene expression. *Proc Natl Acad Sci U S A* 106: 5123–8.
14. Becskei A, S raphin B, Serrano L (2001) Positive feedback in eukaryotic gene networks: cell differentiation by graded to binary response conversion. *EMBO J* 20: 2528–35.
15. Hermesen R, Tans S, ten Wolde PR (2006) Transcriptional regulation by competing transcription factor modules. *PLoS Computational Biology* 2: e164.
16. Buchler NE, Gerland U, Hwa T (2003) On schemes of combinatorial transcription logic. *Proc Natl Acad Sci USA* 100: 5136–5141.
17. Kaplan S, Bren A, Zaslaver A, Dekel E, Alon U (2008) Diverse two-dimensional input functions control bacterial sugar genes. *Mol Cell* 29: 786–792.
18. Collado-Vides J, Magasanik B, Gralla JD (1991) Control site location and transcriptional regulation in *Escherichia coli*. *Microbiol Rev* 55: 371–394.
19. Mandel-Gutfreund Y, Margalit H (1998) Quantitative parameters for amino acid-base interactions: implications for prediction of protein-DNA binding sites. *Nucl Acids Res* 26(10): 2306–2312.
20. Shea MA, Ackers GK (1985) The  $\sigma$  control system of bacteriophage lambda. a physical-chemical model for gene regulation. *J Mol Biol* 181: 211–230.
21. Lissner S, Margalit H (1993) Compilation of *E.coli* mRNA promoter sequences. *Nucl Acids Res* 21(7): 1507–1516.
22. Berg OG, von Hippel PH (1987) Selection of DNA binding sites by regulatory proteins. Statistical-mechanical theory and application to operators and promoters. *J Mol Biol* 193: 723–750.
23. Berg OG, von Hippel PH (1988) Selection of DNA binding sites by regulatory proteins ii. The binding specificity of cyclic AMP receptor protein to recognition sites. *J Mol Biol* 200: 709–723.
24. Berg OG (1988) Selection of DNA binding sites by regulatory proteins. Functional specificity and pseudosite competition. *J Biomol Struct & Dynam* 6(2): 275–297.
25. Ptashne M, Gann A (1997) Transcriptional activation by recruitment. *Nature* 386: 569–577.
26. Bintu L, Buchler NE, Garcia HG, Gerland U, Hwa T, et al. (2005) Transcription regulation by the numbers I: Models. *Curr Opin Gen & Dev* 15: 116–124.
27. Swain PS, Elowitz MB, Siggia ED (2002) Intrinsic and extrinsic contributions to stochasticity in gene expression. *Proc Natl Acad Sci U S A* 99: 12795–800.
28. Elowitz MB, Levine AJ, Siggia ED, Swain PS (2002) Stochastic gene expression in a single cell. *Science* 297: 1183–6.
29. Maeda S, Mizuno T (1990) Evidence for multiple ompR-binding sites in the upstream activation sequence of the ompC promoter in *Escherichia coli*: a single ompR-binding site is capable of activating the promoter. *J Bacteriol* 172: 501–3.
30. Ratna P, Scherrer S, Fleischli C, Becskei A (2009) Synergy of repression and silencing gradients along the chromosome. *J Mol Biol* 387: 826–39.
31. Zinzen RP, Senger K, Levine M, Papatsenko D (2006 Jul 11) Computational models for neurogenic gene expression in the *Drosophila* embryo. *Curr Biol* 16: 1358–1365.
32. Freeman M (2000) Feedback control of intercellular signalling in development. *Nature* 408: 313–9.
33. Kussell E, Leibler S (2005) Phenotypic diversity, population growth, and information in fluctuating environments. *Science* 309: 2075–2078.
34. Tanase-Nicola S, ten Wolde PR (2008) Regulatory control and the costs and benefits of biochemical noise. *PLoS Comput Biol* 4: e1000125.