

Proportional-Integral Control of Stochastic Gene Regulatory Networks

Eric Klavins
Electrical Engineering
University of Washington
Seattle, WA 98195

Abstract—We consider the problem of controlling gene expression in a stochastic genetic regulatory network. We introduce a general class of network designs that achieve this task, and that could be implemented in a synthetic biological setting. For our system, we describe the dynamics of the means and covariances of the gene products, and show that their steady states can be prescribed and tuned. We demonstrate the ideas with a series of examples and also discuss how the network design could be implemented with molecular mechanisms such as transcription factors and kinases.

I. INTRODUCTION

The goal of synthetic biology is to design novel, robust, and tunable biochemical systems and program them into the genes of target cells. From the relatively simple bistable switch [11] and repressilator [9], published a decade ago, to the remarkably sophisticated coupled oscillator published this year [7], incredible progress has been made in our understanding of how to reprogram cells. For control systems, synthetic biology offers a new class of systems with very new challenges. For example, consider implementing (with molecules!) an LQG controller to regulate the concentration of a novel metabolite introduced by a metabolic pathway engineer. The lack of a true system model, the uncertainty in the molecular components of the implementation, the inherent noise inside a single cell [22], [18], and the question of how to represent a state estimate and do matrix multiplication with molecules present enormous hurdles. In fact LQG and many other control design paradigms that work well for micro-controllers are likely impractical for synthetic biology.

However, cells clearly have sophisticated control systems [24], [8]. In fact, what control systems engineers call *regulation* and *zero steady state error*, systems biologists call *homeostasis* and *perfect adaptation* [3]. Control systems inside cells can tightly regulate molecular *copy numbers* in the face of intrinsic and extrinsic noise, even though these control systems themselves are implemented with molecules subject to the same noise sources. The question of how novel control systems can be designed *de novo* for such tasks is the subject of this paper.

In particular, we consider the task of designing a gene regulatory network consisting of n genes that can be tuned so that (1) the mean *copy numbers* of the gene products can be

set arbitrarily; (2) the covariance matrix of the copy numbers of the gene products can be prescribed using a scheme akin to full state feedback.

Because the state of our system consists of copy numbers for some molecular species and concentrations for others, we express the network as a *stochastic hybrid system* [13] and determine the *moment dynamics* of the system to describe the dynamics of the mean and covariance. The enabling idea in the paper is to define the dynamics so that they are linear, so that the moment dynamics are *closed* [15]. This idea was also employed in [23] for the purposes of analyzing noisy gene expression models. As a result, the analysis of the stochastic process that results is straightforward. We use the simple idea of integral control, which we have shown works for copy-number control in a stochastically-interacting robotic setting [20]. Of course, an actual implementation of the scheme in this paper would likely be nonlinear and we discuss one such possible implementation in Section V.

The structure of the paper is as follows. In Section III we define the network structure and analyze its properties. In Section IV we discuss several increasingly complex examples that show how the design works. In Section V, we discuss initial ideas for how the design could be implemented in molecules.

II. PRELIMINARIES

A chemical reaction network in which some species occur in low numbers (e.g. less than one thousand copies) can be modeled using the chemical master equation [12], or CME. The CME involving n chemical species with discrete copy numbers defines a Markov process on the n -dimensional lattice of non-negative integers. Each point in the lattice corresponds to a vector of molecular *copy numbers*.

The system described in this paper, however, also includes species whose numbers are so great as to be better modeled by concentrations. Thus, the state of the system is described by two vectors: A vector $X(t)$ describing the copy numbers of the low-copy species and a vector $Z(t)$ describing the concentrations of the other species. For example, the copy number of some proteins in a single *E. coli* cell can be less than 100 while the copy number of others can be in the millions [1, ch. 2].

Here, we aim to describe the stochastic processes $X(t)$ and $Z(t)$ for a particular network design involving idealized

transcription factors and other molecules (as described in Section III-A). We focus on the dynamics of the expected values of X and Z , which we denote here by $\langle X \rangle$ and $\langle Z \rangle$, as well as the dynamics of the second moments $\langle XX^T \rangle$, $\langle XZ^T \rangle$, and $\langle ZZ^T \rangle$ from which the covariance matrix on X and Z can be obtained. In particular, convergence results in this paper involve convergence in mean and (co)variance.

To simplify notation, we define a vector for the means and a matrix for the second moments

$$\mu = \begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix} \triangleq \begin{pmatrix} \langle X \rangle \\ \langle Z \rangle \end{pmatrix} \quad \text{and} \quad M \triangleq \begin{pmatrix} \langle XX^T \rangle & \langle XZ^T \rangle \\ \langle ZX^T \rangle & \langle ZZ^T \rangle \end{pmatrix}.$$

We also require the covariance matrix and its sub-blocks, defined as

$$\kappa = \begin{pmatrix} \kappa_{XX^T} & \kappa_{XZ^T} \\ \kappa_{ZX^T} & \kappa_{ZZ^T} \end{pmatrix} \triangleq M - \mu\mu^T.$$

Recall that κ is a covariance matrix if and only if it is (symmetric) positive semidefinite.

To write the dynamics of a chemical system with mixed continuous and discrete states, we use a basic result from stochastic hybrid systems [14]. In particular, let ψ be a real-valued function of $X(t)$ and $Z(t)$. Then the dynamics of the expected value of ψ is described by

$$\frac{d}{dt} \langle \psi \rangle = \langle L\psi \rangle$$

where L is the *extended generator*. Its general definition [14] is more than we need here. In the present context,

$$L\psi = \frac{\partial \psi}{\partial Z} f(X, Z) + \sum_i [\psi(\phi_i(X), Z) - \psi(X, Z)] \lambda_i(X, Z).$$

In this expression, $\dot{Z} = f(X, Z)$ describes the continuous dynamics of the system. The sum is over all discrete transitions (changes in copy numbers). That is, $\phi_i(X)$ is the copy number vector obtained from X via the discrete transition i . Also, λ_i , which may be a function of X and Z , describes the rate of the transition. That is, $\lambda_i dt$ is probability that the i th transition occurs in the next dt seconds.

The system proposed in this paper for gene regulation is a linear model that only approximates more realistic models of biochemical dynamics. To validate our results, we compare the linear model to stochastic simulations of more detailed models. For discrete chemical systems, the *stochastic simulation algorithm* [12] would be sufficient for this purpose. However, for the mixed discrete/continuous dynamics defined here, we simply use Euler integration with a small dt .

III. PI CONTROL OF GENE NETWORKS

A. Network Design

We consider a network of n transcription factors. The copy number of transcription factor i is denoted $X_i(t) \in \mathbb{Z}$. In our formulation, each transcription factor i requires an auxiliary *integrator species* with concentration $Z_i(t) \in \mathbb{R}$. We put $X = (X_1 \dots X_n)^T$ and $Z = (Z_1 \dots Z_n)^T$

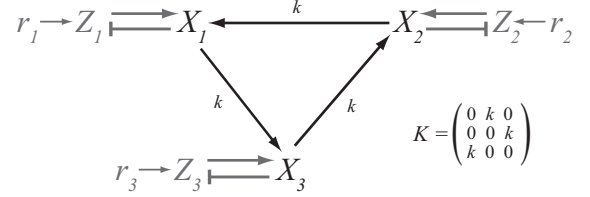


Fig. 1. An example of the design described by (1), (2), and (3) in which the network has a ring structure. Genes (X_i) are controlled by integrators (Z_i) with reference inputs r_i . Genes are connected into a regulator network via proportional feedback terms specified by the matrix K . Note, protein degradation and reference inputs are not shown.

The production and degradation of X_i follows

$$\emptyset \xrightarrow{u_i(X, Z)} X_i \xrightarrow{\beta_i} \emptyset \quad (1)$$

where $\beta_i > 0$ is the (generally uncertain) rate at which X_i is degraded and

$$u_i(X, Z) = \gamma_i Z_i - \sum_{j=1}^n k_{ij} X_j \quad (2)$$

is the rate at which X_i is produced. In (2), γ_i is the *integral gain* and $k_{i,j}$ is the *proportional gain* from X_j to X_i . We define $u = (u_1 \dots u_n)^T$. Note that in this model u_i could be negative, not befitting a proper reaction rate. However, in the systems we explore, the probability that u_i is negative is very small. At this time we do not have a formal justification for this statement. However, when we implement u_i using repressors and activators, it will be impossible for the the actual production rate to be negative.

The dynamics of the integrator species i follows

$$\dot{Z}_i = r_i - X_i \quad (3)$$

where $r_i > 0$ is called the *reference input* for transcription factor i . Once again, Z_i could become negative, but in the systems we explore, it has a low probability of doing so *and* when we implement Z_i using biochemical reactions, negativity becomes impossible.

Collecting the parameters and gains, we define

$$\begin{aligned} P &\triangleq \text{diag}(\beta_1, \dots, \beta_n) \\ \Gamma &\triangleq \text{diag}(\gamma_1, \dots, \gamma_n) \\ K &\triangleq \{k_{ij}\} \\ r &\triangleq (r_1 \dots r_n)^T. \end{aligned}$$

Notice that K essentially defines a graph as shown in Figure 1, with an arrow leading from j to i when $k_{ij} \neq 0$. If k_{ij} is positive, then X_j represses the production of X_i . If $k_{i,j}$ is negative, then it activates the production of X_i . In Section V we describe how (2) could be implemented with appropriately tuned transcription factors.

B. The Moment Dynamics of the Network

The extended generator of the system defined by (1) and (3) is given by

$$\begin{aligned} L\psi &= \frac{\partial \psi}{\partial Z}(r - X) \\ &+ \sum_{i=1}^n [\psi(X + e_i, Z) - \psi(X, Z)] u_i(X, Z) \\ &+ \sum_{i=1}^n [\psi(X - e_i, Z) - \psi(X, Z)] \beta_i X_i \end{aligned}$$

where e_i denotes the i th unit vector.

Applying L to the elements of μ and collecting into vectors results in the dynamics of the first moment:

$$\frac{d}{dt} \begin{pmatrix} \mu_X \\ \mu_Z \end{pmatrix} = \begin{pmatrix} -P - K & \Gamma \\ -I & 0 \end{pmatrix} \begin{pmatrix} \mu_X \\ \mu_Z \end{pmatrix} + \begin{pmatrix} 0 \\ I \end{pmatrix} r, \quad (4)$$

or written more compactly,

$$\dot{\mu} = A\mu + Br$$

where $A \in \mathbb{R}^{2n \times 2n}$ and $B \in \mathbb{R}^{2n \times n}$ have the obvious meanings.

Similarly, applying the extended generator L to the elements of M gives

$$\dot{M} = AM + MA^T + C(\mu) \quad (5)$$

where A is the same as above and $C(\mu) \in \mathbb{R}^{2n \times 2n}$ is the matrix

$$\begin{pmatrix} \text{diag}(\Gamma\mu_Z + (P - K)\mu_X) & \mu_X r^T \\ r \mu_X^T & \mu_Z r^T + r \mu_Z^T \end{pmatrix}.$$

Together, (4) and (5) determine enough of the dynamics of the system for use to reason about the *ensemble dynamics* of the system. For example, from these equations we can reason about the rate of convergence in mean and variance, and the mean and variance of the stationary distribution. Often, this is the same information that can be gleaned from experiments in systems and synthetic biology using fluorescence microscopy [17] or flow cytometry [19].

C. Properties of the Network

Ideally, all the statistics of a synthetic gene network architecture would be robust and tunable. In the present context, we would like to tune the network so that it converges (in mean and variance) and we would like to be able to set the mean μ and covariance κ arbitrarily. Furthermore, we would like these statistics to be robust to a variety of disturbances, for example, disturbances in the rates of production and degradation or unmodeled interactions.

Our first result is that an appropriately designed K and γ result in a stable system. One simply has to check the eigenvalues of A .

Theorem 1: The network converges in mean and variance if and only if A is Hurwitz.

Proof: Clearly (4) is asymptotically stable when A is Hurwitz. Now let $\mathcal{A}(M) = AM + MA^T$. The eigenvalues of the linear operator \mathcal{A} are all possible sums of pairs of

eigenvalues of A and A^T [4, p.71]. Thus, if A has all eigenvalues in the left-half plane, so does \mathcal{A} . ■

It would be nice if the poles of A could be placed arbitrarily, and in small examples we are able to do so by prescribing the coefficients of the characteristic polynomial of A (see Example IV-A). However, A does not separate into the form $\tilde{A} - \tilde{B}\tilde{K}$. In general the coefficients of the characteristic polynomial of A are nonlinear in K and Γ . Thus, prescribing these coefficients is not straightforward.

Prescribing the exact mean in X and covariance matrix in X is possible, however.

Theorem 2: The unique steady state mean μ_X^* is r and is insensitive to K , Γ , and P .

Proof: Set (4) to zero and solve for $\langle X \rangle$. ■

To determine the steady state covariance matrix, we substitute $M = \kappa + \mu\mu^T$ into (5) and set $\dot{M} = 0$ to get

$$A\kappa + \kappa A^T = -C(\mu) - A\mu\mu^T - \mu\mu^T A.$$

Simplifying the right hand side results in a particularly simple Lyapunov equation:

$$A\kappa^* + \kappa^* A^T = \begin{pmatrix} -\text{diag}(2Pr) & 0 \\ 0 & 0 \end{pmatrix}, \quad (6)$$

If A is non-singular (which it is if the system is stable), then the existence of a unique steady state covariance matrix κ^* is guaranteed [4, p. 71]. For purposes of design, we would like to prescribe the covariance matrix and find K and Γ to solve (6). If κ^* is chosen to be positive definite, it would seem that an A exists to solve (6). However, such an A would have to have the form in (4), which is not guaranteed. Also, κ^* is not just any positive definite matrix. For example, (6) implies that κ_{XZ^T} is skew-symmetric (which itself implies that X_i and Z_i are uncorrelated in steady state). Nevertheless, we can show the following theorem.

Theorem 3: The steady state covariance matrix in X can be placed arbitrarily. That is, if W is positive definite, then K and Γ can be found so that in steady state $\kappa_{XX^T}^* = W$.

Proof: We show that we can find K and Γ so that $\kappa_{XX^T}^* = W$, $\kappa_{ZZ^T}^* = \Gamma^{-1}W$, and $\kappa_{XZ^T}^* = 0$. In fact, substituting these assignments into (6) gives

$$-(K + P)W - W(K + P)^T = \text{diag}(2Pr) \quad (7)$$

$$\Gamma\kappa_{ZZ^T}^* - W = 0. \quad (8)$$

Since W is positive definite, there exists a unique gain matrix K solving (7). Since $\kappa_{ZZ^T}^*$ is symmetric, Equation 8 requires that $\Gamma = \gamma I$ for some $\gamma \in \mathbb{R}$. Then $\kappa_{ZZ^T}^* = \gamma^{-1}W$. ■

It is quite surprising that, for our system, steady state covariance matrices can be constructed with arbitrary $\kappa_{XX^T}^*$ even though $\kappa_{XZ^T}^* = 0$. In fact, there are also steady state covariance matrices for the system that have $\kappa_{XZ^T}^* \neq 0$ (when Γ is not a scalar multiple of I), but K is more difficult to find in such cases.

IV. EXAMPLES

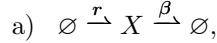
The goal of this section is to survey the possible behaviors and design problems associated with networks defined by (1), (2), and (3). Details regarding the values of the rate

constants and whether they are biologically relevant, or the units used, etc. are beyond the scope of this paper.

A. Control of a Single Gene

In this example, we compare three different architectures for regulating the product of a single gene. The goal of the comparison is to demonstrate which network is the most tunable and least sensitive to unknown parameters. The networks span the range from (a) open loop, (b) proportional control, and (c) proportional-integral control as described in the previous section.

In the first network



the product X is produced and degraded *open loop*. In this case, the extended generator is

$$\begin{aligned} L_a \psi &= [\psi(X+1) - \psi(X)]r \\ &+ [\psi(X-1) - \psi(X)]\beta X \end{aligned}$$

and the dynamics of the first and second moments are

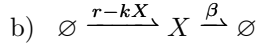
$$\frac{d}{dt} \begin{pmatrix} \langle X \rangle \\ \langle X^2 \rangle \end{pmatrix} = \begin{pmatrix} -\beta & 0 \\ 2r + \beta & -2\beta \end{pmatrix} \begin{pmatrix} \langle X \rangle \\ \langle X^2 \rangle \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} r.$$

For a given production rate r , this system is stable when $\beta > 0$. Putting $\mu = \langle X \rangle$ and $\kappa = \langle X^2 \rangle - \langle X \rangle^2$ and solving the above for steady state gives

$$\mu^* = \frac{r}{\beta} \quad \text{and} \quad \kappa^* = \frac{r}{\beta}. \quad (9)$$

In fact, it can be shown that the steady state is Poisson with parameter r/β . In this network, we can see that the steady state is quite sensitive to the parameters β , and importantly, the variance and the mean are not independently tunable.

In the second network



the product X represses its own production, providing proportional control. The extended generator is

$$\begin{aligned} L_b \psi &= [\psi(X+1) - \psi(X)](r - kX) \\ &+ [\psi(X-1) - \psi(X)]\beta X \end{aligned}$$

and the dynamics of the first and second moments are, similar to the above,

$$\frac{d}{dt} \begin{pmatrix} \langle X \rangle \\ \langle X^2 \rangle \end{pmatrix} = \begin{pmatrix} -\beta - k & 0 \\ 2r + \beta - k & -2\beta - 2k \end{pmatrix} \begin{pmatrix} \langle X \rangle \\ \langle X^2 \rangle \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} r.$$

Assuming $\beta > 0$, this system is stable when $k > -\beta$, and the steady state mean and variance are

$$\mu^* = \frac{r\beta}{(k+\beta)^2} \quad \text{and} \quad \kappa^* = \frac{r}{k+\beta}. \quad (10)$$

In contrast to (9), the mean and variance in (10) are independently tunable (there are two equations and two parameters, k and r). Furthermore, the variance can be decreased by increasing the feedback gain, k . However, to decrease the variance and achieve the same mean, r must be increased as well, and therefore, the mean value remains sensitive to β .

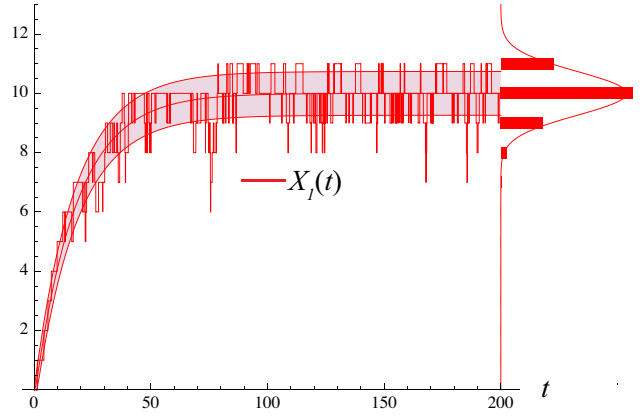
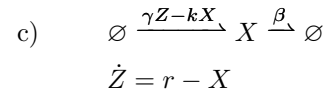


Fig. 2. A simulation of the control of a single gene using proportional-integral control with $r = 10$, $\beta = 0.5$, $k = 2$, and $\gamma = 0.2$. Shown are the predicted mean and one-standard deviation window; a single trajectory from a stochastic simulation of the network that disallows negative rates; a histogram of the last half of the trajectory; and a Gaussian with the predicted steady state mean ($r = 10$) and variance $(\beta r)/(k + \beta) = 2$.

The third network,



designed according to the scheme described in Section III-A, includes an integrator Z . The resulting dynamics follow equations (4) and (5) with

$$A = \begin{pmatrix} -\beta - k & \gamma \\ -1 & 0 \end{pmatrix}.$$

The characteristic polynomial of A is $s^2 + (k + \beta)s + \gamma$, showing that the poles can be placed arbitrarily by k and γ . Furthermore, the steady state mean is simply r and the steady state covariance matrix is

$$C^* = \begin{pmatrix} \frac{r\beta}{k+\beta} & 0 \\ 0 & \frac{r\beta}{(k+\beta)\gamma} \end{pmatrix}. \quad (11)$$

Significantly, the variance in X can be tuned via k without affecting the mean.

Figure 2 shows a simulation of the third network. Overlaid with the moment dynamics is a stochastic simulation of the system in which the production rate is set to $\max(0, \gamma Z - kX)$ to prevent negative rates. The steady state mean and variance are also shown as a histogram and compared to a Gaussian with the predicted mean and variance.

B. Prescribing the Covariance Matrix

As an example design problem, suppose we are given $\beta_1 = 0.1$ and $\beta_2 = 0.5$ and we are tasked with finding K and Γ so that

$$\mu^* = r = \begin{pmatrix} 10 \\ 20 \end{pmatrix} \quad \text{and} \quad \kappa_{XX^T} = \begin{pmatrix} 1 & 2 \\ 2 & 10 \end{pmatrix}.$$

According to Theorem 3, if we are satisfied with having $\kappa_{ZZ^T}^* = 0$, then any symmetric Γ will do. For our example,

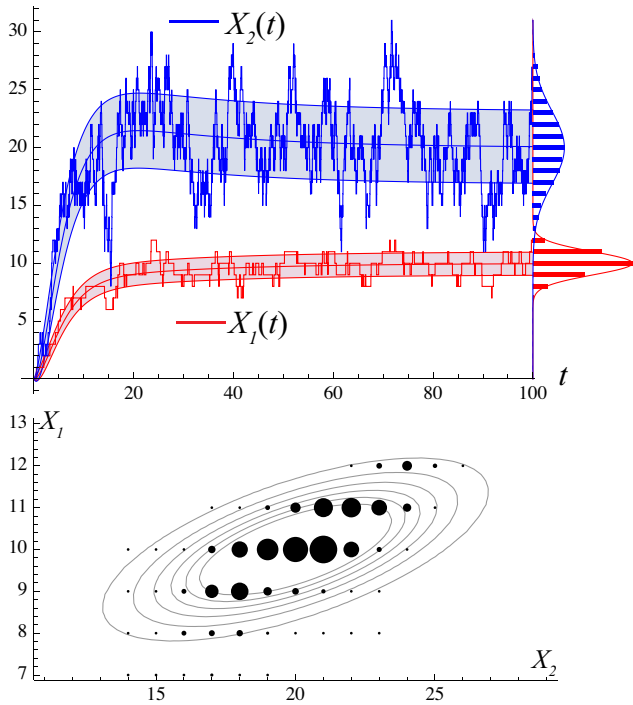


Fig. 3. (Top) A simulation of the control of two single genes using proportional-integral control and a fully connected network. (Bottom) Points from the simulation from $t = 150$ to $t = 300$ (not shown in the top figure). The point size indicates relative frequency. Contour lines are for a Gaussian with the mean and covariance matrix specified in Section IV-B.

we choose $\Gamma = 0.5I$. To find K , we solve (7) with W set according to the above specification and obtain

$$K = \begin{pmatrix} 1.73 & -0.417 \\ -2.5 & 1.0 \end{pmatrix}$$

In fact, the system is underspecified so that there is a one parameter family of solutions. One can check that all of them, together with our choice of Γ make A Hurwitz.

A simulation of (4) and (5) for the above values of K and Γ are shown in Figure 3. A stochastic simulation is overlaid along with the histograms sampled from the steady state (the system is ergodic so sample a window from steady state and taking an ensemble average coincide). Although we have not solved for the steady state distribution for X_i , we approximate the distribution with a Gaussian having the predicted steady state mean and variance. The bottom part of Figure 3 shows that covariance between X_1 and X_2 was obtained.

C. A Three Gene System

The network in Figure 1 has the topology of a repressilator [9] – although not exactly the dynamics of one. If properly tuned, it the eigenvalues of A have significant imaginary parts and the system exhibits damped oscillations. However, the noise inherent in the low copy number aspect of the system constantly excites it, so that even though the ensemble behavior stabilizes to a constant signal, the individual trajectories of the three gene products oscillate 120 degrees out of phase.

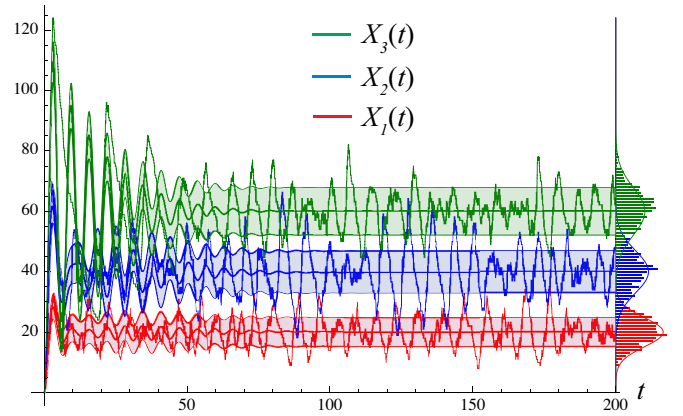


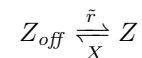
Fig. 4. A simulation of the three gene network shown in Figure 1. The ensemble has the dynamics of a damped oscillator, and eventually settles to a constant signal. The individual trajectories constant excite the oscillatory modes of the system and, therefore, oscillate with the same period as the ensemble, even during steady state. It is not a very good oscillator, but the example does show the possibilities of the class of systems defined in this paper.

Figure 4 shows a simulation of the system in Figure 1 with $k = 0.1$, $\gamma_i = 1$, and $r = (20 \ 40 \ 60)^T$. Note that the reference inputs r_i can be used to shift the means of the individual gene products arbitrarily.

V. IMPLEMENTATION WITH BIOCHEMICAL REACTIONS

In this section we discuss the plausibility of implementing the idealized dynamics described above via actual biochemical mechanisms. It should not be surprising that biochemistry could do these reactions. Several authors have shown that biochemistry can implement Boolean logic [10], linear systems [21], and Turing complete computation [5]. Therefore, here we simply describe some ideas for how we might proceed. The actual mechanisms would have to be carefully chosen and tuned, of course. Based on our own work in synthetic biology, we believe that the mechanisms discussed below could be made to work, in for example, *E. coli*. But this is just a conjecture.

As discussed above, many biochemical implementations of what appear to be integrators have been elucidated in the systems biology literature. One way to make Z integrate a difference as in (3) is with a reaction of the form



where Z_{off} is an inactive form of the transcription factor Z . A small molecule presumably with concentration \tilde{r} , perhaps a metabolite or attractant, activates the transcription factor, putting it into its active form Z . If Z_{off} is regulated and in large supply, the rate of this reaction is \tilde{Z}_{off} , which we call r . For example, the small molecule may recruit a kinase that phosphorylates Z_{off} to obtain Z (there are examples of this in glycolysis [2]). In the reverse reaction, X deactivates, perhaps dephosphorylates, Z . Of course, X is also a transcription factor. One idea is to make a protein fusion of a transcription factor and a phosphatase. Considerably more unlikely pairs are regularly fused. As for the rate of the

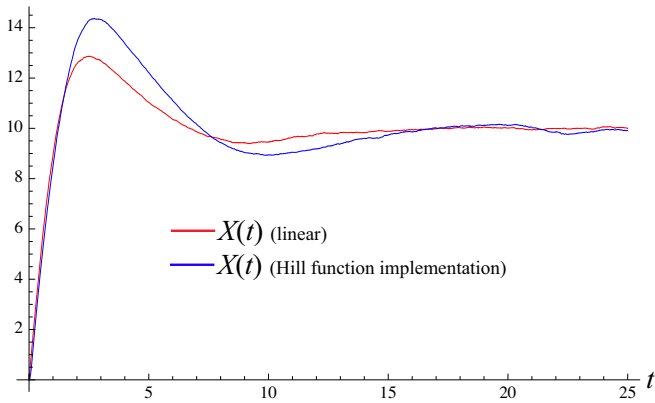


Fig. 5. The average of 1000 simulations of a one gene network (red) and 1000 simulations of the network implemented with the Hill function (12) (blue). The parameters of the Hill function were $n = m = K_X = 1$, $K_Z = 100$ and $v = 1331$, which gives $h(X, Z) \approx 11 + z - x$.

reaction, it would seem to be proportional to XZ . However, it is likely that the activity of the enzymatic activity of X could be tuned to operate in saturation. Since X is low copy and Z is high copy, that seems reasonable.

As for implementing (2), consider the problem of implementing the third network in Section IV-A. We need to have X repress gene expression and Z activate it. A promoter that can be both activated and repressed can readily be found, for example, from the combinatorial promoter library [6]. Such a dual promoter would result in a rate of gene expression in the form of a Hill function [16, p. 268]:

$$h(X, Z) = \frac{vZ^m}{(K_Z + Z^m)(K_X + Z^n)} \quad (12)$$

where v , n , m , K_Z and K_X are parameters that can be tuned by modifying promoter strengths, ribosome binding site affinities, and so on. Linearizing h near μ_X^* and μ_Z^* gives

$$\begin{aligned} h(X, Z) &\approx h(\mu_X^*, \mu_Z^*) \\ &+ \left. \frac{\partial h(X, Z)}{\partial X} \right|_{\mu=\mu^*} (X - \mu_X^*) \\ &+ \left. \frac{\partial h(X, Z)}{\partial Z} \right|_{\mu=\mu^*} (Z - \mu_Z^*) \\ &= \alpha + \gamma Z - kX. \end{aligned}$$

Thus, one tunes the parameters in h to give the desired γ and k . The constant term α does not appear in (2). Adding it does not substantially change the analysis in this paper, however.

Figure 5 shows a the average of 1000 stochastic simulations of a one-gene network (red) with the production rate equal to $11 + z - x$. It also shows the average of 1000 stochastic simulation in which we implement the production rate with a Hill function $h(X, Z)$ that approximates $11 + z - x$. The steady states are the same (as expected) and the transients are also quite similar.

VI. DISCUSSION AND EXTENSIONS

The system described by (1), (2), and (3) is an example of a network design that arguably could be implemented in cells. It is easy to solve a number of design problems in this setting, from placing poles to prescribing stationary distributions. In contrast, understanding the dynamics of existing biochemical networks (as is the task of the systems biologist) can be quite difficult. Evolution optimizes performance (broadly construed), not understandability.

Many questions remain, several of which we highlight here. (a) The network structure itself can be altered and adapted to specific settings. For example, if the task is to control the concentration of a particular high-concentration metabolite using low-copy-number enzymes, what network structure should be used? (b) The network could include more details. RNA dynamics were not modeled. Cooperativity and protein-protein interactions were not modeled. It should be emphasized, however, that increasing the fidelity of the model is only necessary in as much as it may introduce new kinds of dynamics or new knobs to turn. As a specification of a desired system, simplicity is crucial and other details should be left to a lower level of abstraction or implementation. (c) Several design problems requiring more sophisticated treatments than in this paper seem like obvious next steps. Can K and Γ be found to satisfy a given performance criterion given constraints on the network connectivity (e.g. find the sparsest K that minimizes $\|W - \kappa_{XX}^*\|$)? What can be achieved given biologically realistic limitations on the parameters? (d) The interaction between the concentration of Z and the copy number of X seems to be required. If Z is implemented in high copy number, but not so high as to be modeled as an element of \mathbb{R} , what limitations result? (e) And finally, can this system (or similar) be implemented and easily tuned? Many of the tools are in place to investigate possible implementations and much of the current work in the author's lab is in this direction.

REFERENCES

- [1] U. Alon. *An Introduction to Systems Biology: Design principles of Biological Circuits*. Chapman & Hall/CRC, 2007.
- [2] C. Bächler, P. Schneider, P. Bähler, A. Lustig, and B. Erni. *Escherichia coli* dihydroxyacetone kinase controls gene expression by binding to transcription factor DhaR. *The EMBO journal*, 24(2):283–293, 2005.
- [3] N. Barkai and S. Leibler. Robustness in simple biochemical networks. *Nature*, 387:913–917, 1997.
- [4] C.-T. Chen. *Linear System Theory and Design*. Oxford University Press, 3 edition, 1998.
- [5] M. Cook, D. Soloveichik, E. Winfree, and J. Bruck. Programmability of chemical reaction networks. In A. Condom, D. Harel, J.N. Kok, A. Salomaa, and E. Winfree, editors, *Algorithmic Bioprocesses*, Natural Computing Series, pages 543–584. Springer, 2010.
- [6] R.S. Cox, M.G. Surette, and M.B. Elowitz. Programming gene expression with combinatorial promoters. *Molecular Systems Biology*, 3(145), 2007.
- [7] T. Danino, O. Mondragón-Palomino, L. Tsimring, and J. Hasty. A synchronized quorum of genetic clocks. *Nature*, 463:326–330, 2010.
- [8] H. El-Samad, H. Kurata, J.C. Doyle, C.A., and M. Khamash. Surviving heat shock: Control strategies for robustness and performance. *Proceedings of the National Academy of Sciences*, 102(8):2736–2741, 2005.

- [9] M.B. Elowitz and S. Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403:335–338, 2000.
- [10] D.Y. Zhang G. Seelig, D. Soloveichik and E. Winfree. Enzyme-free nucleic acid logic circuits. *Science*, 314:1585–1588, 2006.
- [11] T.S. Gardner, C.R. Cantor, and J.J. Collins. Construction of a genetic toggle switch in *Escherichia coli*. *Nature*, 403:339–342, 2000.
- [12] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry*, 81:2340–2361, 1977.
- [13] J. Hespanha and A. Singh. Stochastic models for chemically reacting systems using polynomial stochastic hybrid systems. *International Journal on Robust Control*, 15(15):669–689, Sep 2005.
- [14] João Pedro Hespanha. Polynomial stochastic hybrid systems. In Manfred Morari and Lothar Thiele, editors, *Hybrid Systems: Computation and Control*, number 3414, pages 322–338. Springer-Verlag, Berlin, March 2005.
- [15] João Pedro Hespanha. Moment closure for biochemical networks. In *Proc. of the Third Int. Symp. on Control, Communications and Signal Processing*, March 2008.
- [16] E. Klipp, R. Herwig, A. Kowalk, C. Wierling, and H. Lehrach. *Systems Biology in Practice: Concepts, Implementation and Application*. Wiley, 2006.
- [17] J.C.W. Locke and M.B. Elowitz. Using movies to analyse gene circuit dynamics in single cells. *Nature Review Microscopy*, 7(5):383–392, 2009.
- [18] J. T. Mettetal and A. van Oudenaarden. Necessary noise (review). *Science*, 317(5837):463–464, July 2007.
- [19] B. Munsky, B. Trinh, and M. Khammash. Listening to the noise: Random fluctuations reveal gene network parameters. *Molecular Systems Biology*, 5(318), 2009.
- [20] N. Napp, S. Burden, and E. Klavins. Setpoint regulation for stochastically interacting robots. In *Robotics: Science and Systems V*. MIT Press, 2009.
- [21] K. Oishi and E. Klavins. A biomolecular implementation of linear I/O systems. Submitted.
- [22] N. Rosenfeld, J.W. Young, U. Alon, P.S. Swain, and M.B. Elowitz. Gene regulation at the single-cell level. *Science*, 307(5717):1962–1965, March 2005.
- [23] A. Singh and J. Hespanha. Reducing noise through translational control in an auto-regulatory gene network. In *American Control Conference*, 2009.
- [24] T.M. Yi, Y. Huang, M.I. Simon, and J.C. Doyle. Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proceedings of the National Academy of Sciences*, 97(9):4649–4653, 2000.