Computational design of RNA-based oscillatory circuits

J. Binysh
* *University of Warwick, Complexity Department*

***Abstract*—For the synthetic biologist looking to engineer regulation of genetic circuitry, RNA is an attractive tool, due to the ease of predicting its behaviour from physiochemical models. In this report, we introduce a system of ODE's to model a recently designed synthetic regulatory circuit, before attempting to estimate its unknown parameters using recent time series data. We provide estimates for a subset of parameters, but find many of them to be inestimable. We discuss why this is this case for our system, and suggest how this problem might be resolved.**

I. Introduction

The process of gene expression can be summarised as follows: DNA is read, and a copy of it is made, in the form of an RNA molecule (this is called *transcription*). This RNA molecule (known as messenger RNA, or mRNA) makes its way to a piece of cellular machinery called the Ribosome, which reads it, and makes a protein - which protein is made depends on the DNA sequence originally read (*translation*).

The path from genetic transcription to protein expression is naturally regulated in many ways [1]. This regulation allows the cell to control protein expression, and so cell behaviour, in response to various environmental cues. The natural cell machinery which performs it takes the form of genetic circuits - networks of interacting gene expression regulators. This genetic circuitry offers rich possibilities for modification, and an important goal within synthetic biology is to understand and manipulate it.

As well as acting as the intermediate between DNA and protein, RNA molecules play direct and important roles in regulating gene expression [2]. For the synthetic biologist looking to engineer regulation of genetic circuitry, RNA offers an attractive alternative to more traditional methods, which typically involve using proteins to regulate DNA transcription. In comparison to proteins, it is relatively straightforward to predict the structure and function of an RNA from its sequence using physiochemical models. Recently, this has been exploited to computationally design synthetic sRNA's - small RNA's which do not code for a protein, but rather have some direct regulatory function - with regulatory behaviour that can be predicted [3] [4].

This report will focus on one such sRNA system, introduced in [4]. It will extend existing understanding of it beyond the qualitative by first proposing a quantitative model of its behaviour in the form of a set of ODE's, and then fitting this ODE model to available time series data to estimate its unknown parameters.

The report is structured as follows. In the remainder of this section we review the regulatory system we will consider, and discuss recent single cell fluorescence experiments performed on it. In section II we introduce a set of ODE's to model the system. In section III we attempt to estimate its unknown parameters by fitting the model to time series data. Finally, in section IV, we conclude, and suggest directions for further work.

A. The sRNA regulatory system

In bacteria, one mechanism by which gene expression is naturally regulated is as follows [5]: In order for a bacterial mRNA to be translated into a protein, the Ribosome must initially bind to the mRNA (Fig. 2). This occurs at the Ribosome Binding Site (RBS) [6], a specific nucleotide sequence found on the mRNA.

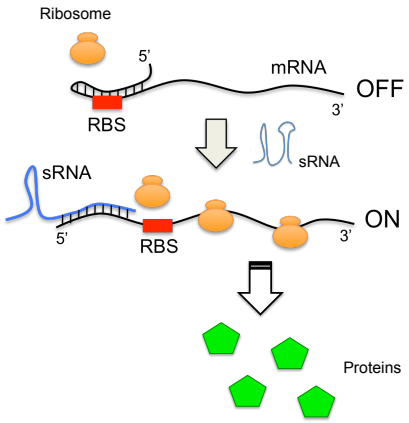


Fig. 2: A mechanism by which sRNA can regulate gene expression. Initially, the 5' UTR of the mRNA is folded over the RBS, forming a loop and blocking Ribosome binding. The sRNA binds to this loop in the mRNA, causing a conformational change which uncovers the RBS, and allows translation to occur. Image reproduced from [4].

In an mRNA there is an untranslated region of nucleotides at the 5' end of the molecule (the UTR), upstream of the RBS [1]. Translation may be self repressed by this 'tail' folding over and binding across the RBS, forming a loop in the mRNA and preventing the Ribosome from binding (Fig. 2). This self repression may be released with an sRNA which binds to

[1]Both DNA and RNA are directional - the backbone of the molecule is not symmetric. This gives the molecule two distinct ends, denoted 5' and 3'. Translation can only occur in the 5' to 3' direction, hence the meaning of 'upstream'

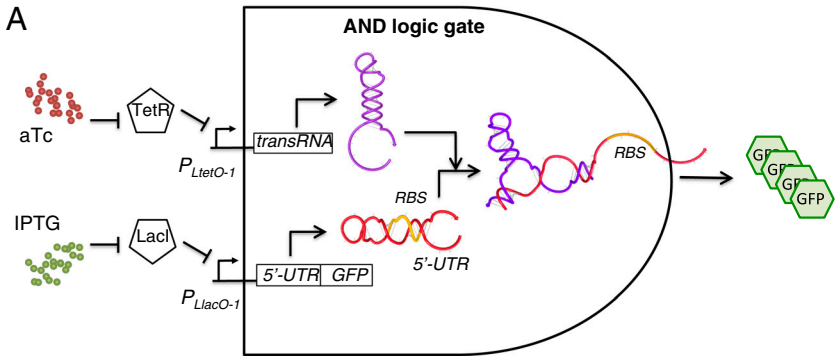


Fig. 1: A logical AND gate formed from a self repressed mRNA, and an sRNA which uncovers its RBS. In this system, transcription of the sRNA (transRNA) and mRNA (5'-UTR,GFP) are controlled by two promoter regions, $P_{\mathrm{LtetO}-1}$ and $P_{\mathrm{LlacO}-1}$. These are disabled by the presence of two chemical repressors, TetR and LacI, found naturally in the strain of *E. coli* discussed. These chemical repressors are themselves disabled by two chemicals, aTC and IPTG. In the notation of the diagram, a barred line indicates repression, and an arrowed line indicates production. We see a 'double negative' in aTc repressing TetR, which itself represses transcription of the sRNA (likewise for IPTG and the mRNA). Thus the presence of the sRNA and mRNA are controlled by the presence of aTc and IPTG, which can be experimentally introduced to the cell. Image reproduced from [4].

this looped tail - the new conformation of the sRNA:mRNA complex uncovers the RBS, allowing the Ribosome to bind. In summary, the presence of the sRNA positively regulates gene expresssion.

[4] proposed a computational methodology to design general genetic circuits based on RNA interactions, and as a case study of the methodology chose to design a synthetic sRNA-mRNA pair capable of acting in the manner described above. The algorithm assumed an interaction scheme between the RNA's as shown in Fig. 3. The sRNA and mRNA, originally in their own individually folded states, would initially interact via a small 'toehold' sequence of unpaired nucleotides to form an unstable transition state. This intermediate complex would then rapidly form a final, stable complex with the desired conformation. By finding sRNA and mRNA sequences which optimised the energy landscape shown in Fig. 3, [4] suggested several sRNA-mRNA pairs which would work in tandem to form a stable hybrid with the RBS free.

The authors then experimentally validated their methodology by testing the suggested sRNA-mRNA pairs in *E. coli*. Further, by placing the *in vivo* concentrations of the sRNA and mRNA under the control of tuneable promoter regions [2], they constructed a logical AND gate from one of the proposed pairs (Fig. 1).

In this system, transcription of the designed sRNA and mRNA are placed under the control of promoter regions, $P_{\mathrm{LtetO}-1}$ and $P_{\mathrm{LlacO}-1}$ [7]. These are in turn controlled by two transcriptional repressors, TetR and LacI, which are naturally present in the strain of *E. coli* considered. These repressors disable the promoter regions, and so by default

[2] A promoter region in a DNA sequence is a sequence of nucleotides, found upstream of where transcription of a gene begins, which can influence the transcription rate of the gene.

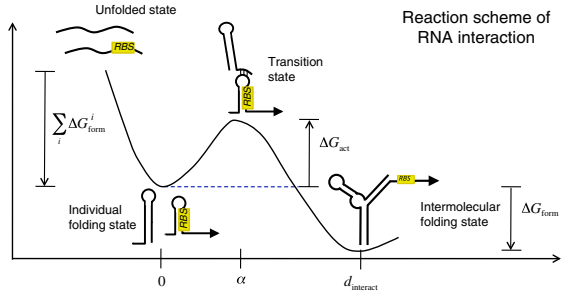
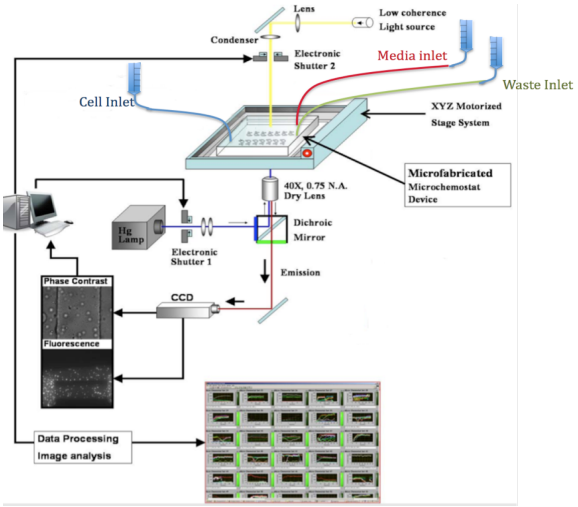


Fig. 3: A proposed reaction scheme between the synthetic sRNA-mRNA pairs designed in [4]. The reaction co-ordinate is defined as the number of paired nucleotides, and the vertical axis denotes free energy. The RNA's initially interact via a small 'toehold' sequence, forming an unstable transition state, which then stabilises to give the final compound. Image reproduced from [4]

transcription of both RNA's is turned off, and no protein is produced. These repressors can themselves be disabled by the presence of two chemicals, aTc and IPTG, which can be introduced externally into the cell (Fig. 1). So transcription of the two RNA's is indirectly controlled by the presence of two chemicals - if neither is present, sRNA and mRNA transcription is repressed, and no protein is produced. If only one is present, the AND gate remains off, either because there is no mRNA to be translated into protein, or because the mRNA is self repressed. But when both are present, the conformational change discussed above occurs, and protein is produced.

Although a qualitative understanding of this system exists [4], it is of interest to attempt a quantitative understanding



Adapted from Balagadde, 2007

Fig. 4: An overview of the experimental setup which allows single cell fluorescences to be recorded over time. Shown are the bacterial growth chambers (labelled 'Microfabricated Microchemstat device'), the microscope imaging them, and the software constructing fluorescence time series. Image reproduced from [8].

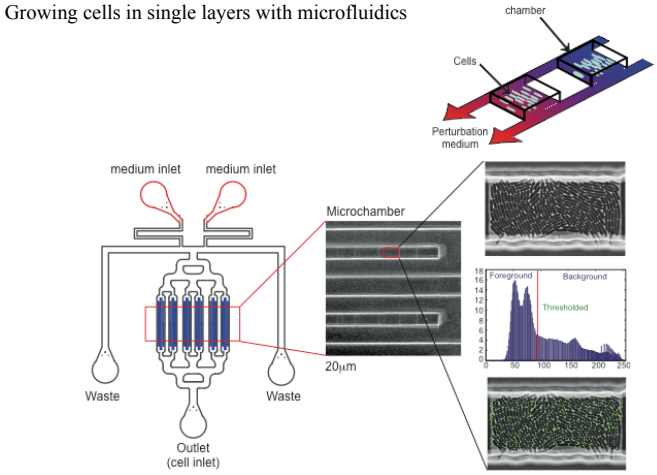


Fig. 5: A diagram of the bacterial growth chambers shown in Fig. 4. The schematic shows the chambers themselves, and the medium inlet where aTc and IPTG are let in. Shown also are photographs of a row of chambers, and an individual chamber with bacteria growing in it. Image reproduced from [8].

of the genetic circuit involved. Such an understanding would allow, for example, tailoring of the system in reponse to design requirements, by altering the values of the important parameters of the model. By changing which sRNA-mRNA pair is used in the system, it would also allow exploration of the relationship between the thermodynamic properties of each device, and the model's rate constants.

B. Single Cell Fluorescence Data

mRNA concentrations in the system shown in Fig. 1 can be indirectly observed by designing the mRNA to code for GFP [3]. Recent experiments have used timelapse microscopy to observe the fluoresence of bacteria which contain the above sRNA-mRNA pair, as they are periodically forced with a varying aTc or IPTG concentration [8].

The experimental setup is as follows (Figs. 4, 5): A single layer of the bacteria are grown in rows of chambers. A medium constantly flows through these chambers, allowing normal feeding of the bacteria, and the introduction of aTc or IPTG. The chambers are monitored with software which traces the position of each cell over time, allowing time series of individual cell fluorescences to be recorded.

The data we will consider consists of two sets of individual cell time series, labelled *13_9* and *14_7*. They correspond to different experimental runs of the above apparatus, for which IPTG concentration was held constant, at a value assumed large enough to saturate the cell's response, and aTc concentrations were varied periodically. Appendix A shows the full datasets, with their forcing functions. Note that *13_9* contains two different forcing periods.

[3]The methodology of [4] only optimises the 5' UTR of the mRNA, so the actual protein being coded for is unimportant.

II. ODE Model

In (1) - (7), we present a modified version an existing model which describes the system, consisting of a set of ODE's with mass action kinetics [9]. Its state is given by the vector $(s, m, s:m, c, p, g, z)$, with all other variables representing model parameters. Tables I and II give complete descriptions of the parameters and state variables.

give the original model and discuss modifications in appendix.

$$\frac{ds}{dt} = \frac{N\alpha_T}{f_T}y(t) - (\mu + \delta_s)s - k_{\text{on}}sm + k_{\text{off}}s:m \quad (1)$$

$$\frac{dm}{dt} = \frac{N\alpha_L}{f_L}x(t) - (\mu + \delta_m)m - k_{\text{on}}sm + k_{\text{off}}s:m \quad (2)$$

$$\frac{ds:m}{dt} = k_{\text{on}}sm - (k_{\text{off}} + k_{\text{hyb}})s:m - (\mu + \delta_{sm})s:m \quad (3)$$

$$\frac{dc}{dt} = k_{\text{hyb}}s:m - (\mu + \delta_c)c \quad (4)$$

$$\frac{dp}{dt} = \beta m + f_s\beta c - (\gamma + \mu + \delta_g)p - \frac{v_z p}{K_z + p + g} \quad (5)$$

$$\frac{dg}{dt} = \gamma p - (\mu + \delta_g)g - \frac{v_z g}{K_z + p + g} \quad (6)$$

$$z = z_0 + \frac{g}{\Theta} \quad (7)$$

Based on the reaction mechanism in Fig. 3, the hybridization of the sRNA and mRNA first into an unstable complex, then

a stable one, is modelled in (1) - (4). The initial binding is modelled as a reversible reaction with forward and backward rates $k_{\rm on}$ and $k_{\rm off}$:

$$\mathrm{sRNA} + \mathrm{mRNA} \underset{k_{\rm off}}{\overset{k_{\rm on}}{\rightleftharpoons}} \mathrm{sRNA:mRNA_{unstable}},$$

after which the stabilization is modelled as an irreversible reaction with rate $k_{\rm hyb}$:

$$\mathrm{sRNA:mRNA_{unstable}} \xrightarrow{k_{\rm hyb}} \mathrm{sRNA:mRNA_{stable}}.$$

In addition, these complexes are given degradation rates, δ_s, δ_m, δ_{sm}, δ_c, and dilutions of chemical concentrations due to cell growth are modelled with a dilution rate μ.

Control of the system by aTc is modelled by $y(t)$ in (1). This function models the response of the sRNA transcription rate to a time varying aTc concentration - it is normalised to lie between 1 and f_T, and is typically sigmoid in response to aTc concentration [4]. α_T is the maximal transcription rate of the $P_{\rm LtetO-1}$ promoter, and N is the copy number, which models the fact that when engineering the system, many copies of the $P_{\rm LtetO-1}$ promoter may be placed in the bacterial DNA. Thus the transcription rate varies as a sigmoid bounded by $N\frac{1}{f_T}$ and $N\frac{\alpha_T}{f}$. Identical considerations hold for $x(t)$ in (2), and IPTG concentration.

We explicitly model translation as a simple one step process in (5) - (7). There is a small rate of translation of the self repressed mRNA [4], which is modelled at rate β, and a larger one for translation of the stable complex, βf_s. Here f_s represents the fractional change in translation rate between the repressed mRNA and the unrepressed complex:

$$\mathrm{sRNA:mRNA_{Stable}} \xrightarrow{\beta f_s} \mathrm{GFP_{Immature}},$$

$$\mathrm{mRNA} \xrightarrow{\beta} \mathrm{GFP_{Immature}}.$$

Initially, the translated GFP is in an immature state, and will not fluoresce. To account for this, we include a maturation rate, γ:

$$\mathrm{GFP_{Immature}} \xrightarrow{\gamma} \mathrm{GFP_{Mature}}.$$

Degradation of the immature and mature GFP is modelled in two ways. Firstly a generic degradation rate δ_g is included, assumed identical for the mature and immature species, along with the dilution rate μ shared by all species. Secondly, in the experimental setup we can arrange for GFP molecules to be produced with a *degradation tag* attached to them [10]. This tag is sought out by a protein, ClpX, which will then degrade the molecule the tag is attached to. This degradation process is modelled by the final terms in (5) and (6). Finally, (7) simply represents calibration of mature GFP levels to experimentally observed fluorescence, assuming a linear response.

TABLE I: State Variables

State variable	Units	**Definition**
s	nM	sRNA concentration
m	nM	mRNA concentration
$s:m$	nM	Unstable sRNA:mRNA complex concentration
c	nM	Stable sRNA:mRNA complex concentration
p	nM	Immature GFP concentration
g	nm	Mature GFP concentration
z	AFU	Observed fluorescence
$y(t)$		Unitless aTc forcing function
$x(t)$		Unitless IPTG forcing function

TABLE II: Model Parameters (those to be estimated shown in bold)

Parameter	**Units**	**Definition**
N		Number of copies of promoter existing on plasmid DNA
z_0	Arbitrary (AU)	Baseline experimental fluorescence
α_L	nM/min	Maximal transcription rate of $P_{\rm LlacO-1}$ promoter
α_T	nM/min	Maximal transcription rate of $P_{\rm LtetO-1}$ promoter
f_L		Unitless ratio between repressed and unrepressed $P_{\rm LlacO-1}$ transcription rate
f_T		Unitless ratio between repressed and unrepressed $P_{\rm LtetO-1}$ transcription rate
δ_g	/min	GFP degradation rate
γ	/min	GFP maturation rate
v_z	nM/min	Degradation constant of clpx
K_z	nM/min	Dissociation constant of clpx
$\boldsymbol{\Theta}$	nM/AFU	Ratio between GFP concentration and observed fluoresence
$\boldsymbol{\mu}$	/min	Dilution rate
$\boldsymbol{\delta_m}$	/min	mRNA degredation rate
$\boldsymbol{\delta_s}$	/min	sRNA degredation rate
δ_{sm}	/min	Unstable sRNA:mRNA degradation rate
δ_c	/min	Stable sRNA:mRNA degradation rate
$\boldsymbol{k_{on}}$	/min	sRNA:mRNA binding rate
$\boldsymbol{k_{off}}$	/min	sRNA:mRNA unbinding rate
$\boldsymbol{k_{hyb}}$	/min	sRNA:mRNA hybridization rate
$\boldsymbol{\beta}$	/min	Baseline translation rate of repressed mRNA
$\boldsymbol{f_s}$		Ratio of repressed mRNA to unrepressed complex translation rate.

III. Parameter Estimation

Our next goal is to estimate the unknown parameters of this model, given the available fluorescence time series data, by fitting the predicted time series from the model to the data. Typically, this is done by minimising the least squares error between model prediction and the experimental data [11]–[13]. Suppose we have some ODE model of our system

$$\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}t} = \mathbf{f}(\mathbf{y}, \boldsymbol{\theta}, t), \tag{8}$$

where $\mathbf{y}$ is our state vector, $\boldsymbol{\theta}$ is a vector of model parameters, and t is time. The model may be integrated numerically, giving a prediction $\mathbf{y}(t, \boldsymbol{\theta})$. The least squares error between the model

prediction and an experimental time series is defined as

$$J(\boldsymbol{\theta}) = \sum_{i=1}^{n} (\mathbf{y}_{\exp}(t_i) - \mathbf{y}(t_i, \boldsymbol{\theta}))^2, \tag{9}$$

where the experimental time series, $\mathbf{y}_{\exp}(t_i)$ is recorded at timepoints t_i, $i = 1 \dots n$. This error function defines a landscape in $\boldsymbol{\theta}$ space, and we seek to minimise it by varying $\boldsymbol{\theta}$. In our case, we do not have experimental data on the full state vector, but only one component of it - the observed fluorescence, $z(t)$. In addition, rather than a single experimental run, we have many, corresponding to a time series from each cell. We incorporate this by fitting to the experimental mean of the data, and only minimising over the observed component. Our minimisation problem is thus

explain why not fit each curve individually.

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (z_{\exp,\text{mean}}(t_i) - z(t_i, \boldsymbol{\theta}))^2. \tag{10}$$

The next step is performing the minimisation. In general, the landscape defined by the error function may be rugged and contain many local minima, which a local optimisation algorithm will may get stuck in [4]. To try and surmount this problem, [12] suggests the use of a global optimisation algorithm, and in particular recommends several Evolutionary Algorithms, of which we choose one, the CMA-ES [14], [15]. [5]

why use CMA-ES? Some refs

In order to reduce the dimensionality of our search space, we can perform a literature search for existing values of some of our parameters, simplify our model to remove others, and place bounds on those that remain. Appendix B contains a list of parameter values found in the literature, where available, and their reference, as well as initial bounds placed on parameters not found in the literature. To further reduce the search space, we simplify the model by assuming that δ_m, δ_{sm} and δ_c all take similar values, and set them equal. After this is done, we are left with a 9 dimensional search space, bounded by a hypercube (parameters to be estimated are shown in bold in table II).

A. Initial Parameter Estimates

We begin by fitting each dataset individually, by choosing 200 parameter sets uniformly distributed over our initial parameter bounds and running the CMA-ES starting from them. Results are shown in Figs. 6, D.1. Figs. 6e, 6f demonstrate that the model is capable of quantitatively capturing the data. However, histograms of estimated parameter values found from fitting the above 200 sets indicate many parameter estimates are not tightly constrained, taking values right across the initial bounding range specified - in particular, k_{off}, δ_m and δ_s are very weakly constrained, with substantial numbers of runs constrained only by the initial bounding box. By contrast, some parameter estimates (μ in particular) are much tighter. Θ appears tightly constrained, but many of the estimated values are close to the upper bound set, and must be treated with caution.

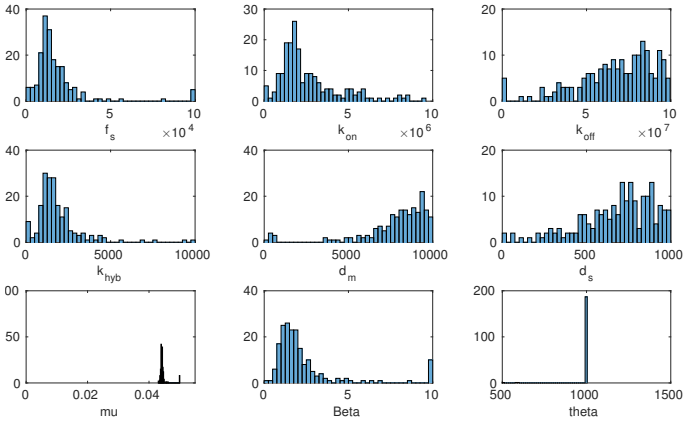
Figs. 6c, 6d show correlation matrices between parameter sets, computed by viewing each of the estimated parameter sets as a sample from a multivariate random variable. An element of the correlation matrix R_{ij} is then the correlation co-efficient between the i^{th} and j^{th} random variable - or, in our case, the i^{th} and j^{th} parameters. These correlation matrices indicate that there are spaces of parameters within which the fitness function remains approximately constant - for example, in both datasets there exists of positive correlation between values of μ and β. Referring to (5), this makes heuristic sense - the two parameters may have similar effects on model predictions, and may be able to co-vary in such a way as to leave the model prediction unchanged. Biologically, an increased production rate of GFP is being balance by an increased dilution rate. The results suggest a fitness landscape relatively flat to perturbations in certain combinations of parameters, and indicate we may have difficulty obtaining unique estimates of the model parameters.

We can test the predictive ability of our model by cross validating, either by taking the parameter values found in the fitting of one dataset and using them to give model predictions for another, or only fitting part of a single dataset and predicting the rest. We begin by fitting to only the data for the first forcing period in *13_9*, and then predicting the full time series. Results are shown in Fig. 7a, for the parameter set giving the lowest error on the training data. We see the prediction is very similar to that obtained by fitting to the full dataset (Fig. 6e). Next, we take the parameter values giving the best fit for the *13_9* dataset and use them to predict the *14_7* dataset, and vice versa. Results are shown in Figs. 7b, 7c. Though the predictions are reasonable, they are substantially worse than the predictions for the data they were trained on. One reason for this may be that parameter values will vary between experimental runs [13]. We can also fit to both datasets combined, by extending (10) to a sum over multiple datasets. Results are shown in Appendix D. We are unable to achieve error values as low as those found when fitting individual datasets. In addition, we do not see a tightening of the estimates on our parameters, as we might expect from including the additional data. Instead, many parameter estimates remain spread across the initial bounding region. We also note that all three of the cases considered (*13_9*, *14_7*, both) see a shift in the estimated value of μ.

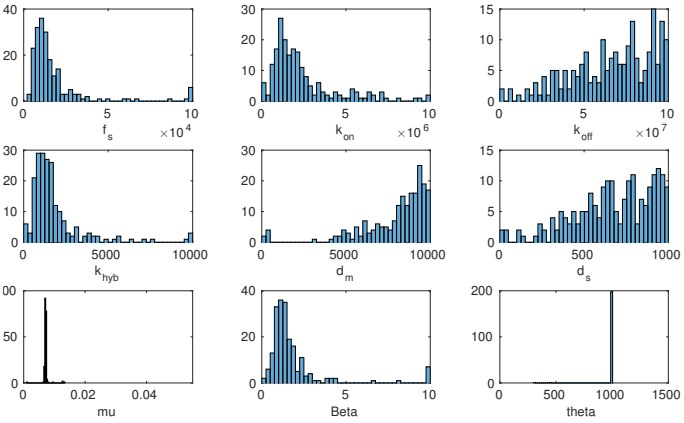
Taken together, these results suggest that simply performing a least squares fit on the available data will not give unique parameter estimates. In the following sections we investigate why this might be.

[4] this is true even if ODE model is linear in its parameters, as is almost the case for us - though the ODE model is linear, the resulting solutions are in general not. A counterexample is the harmonic oscillator.

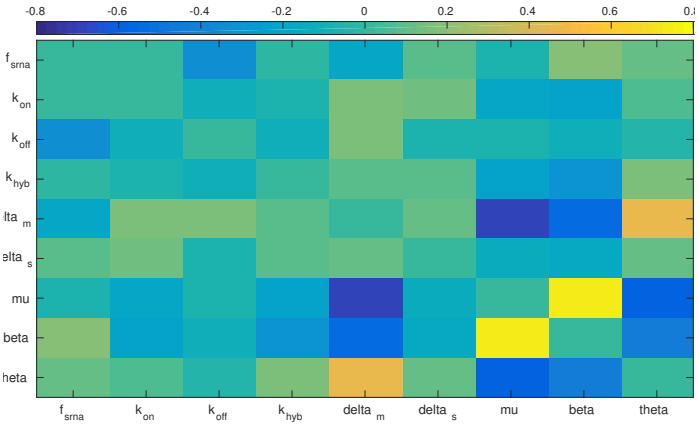
[5] CMA-ES stands for covariance matrix evolutionary strategy. This algorithm generates a population of test points according to a multivariate gaussian distribution. It then ranks the points based on their fitness function scores, and considers some number of the highest ranked points as a sample from a new multivariate gaussian. Using this sample, it then estimates a new mean and covariance matrix for this gaussian and the process is iterated.



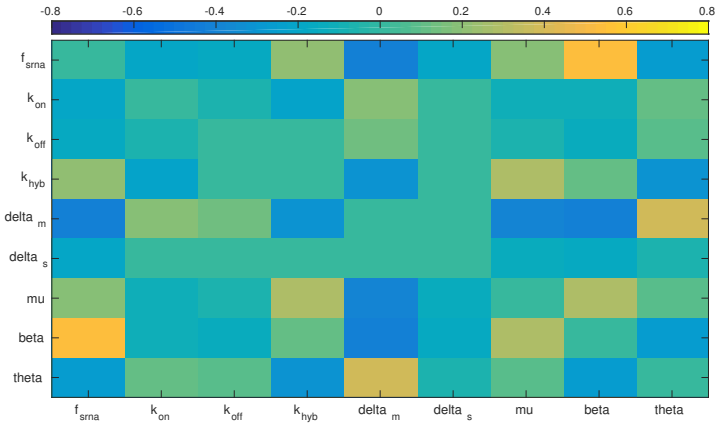
(a) Histogram of estimated parameter values, found from 200 runs of the CMA-ES algorithm. Fitted to the *13_9* dataset.



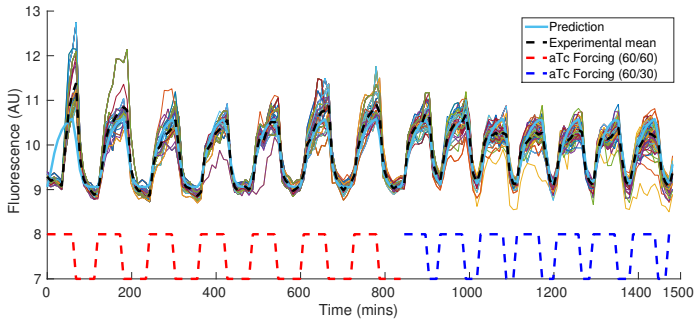
(b) Histogram of estimated parameter values, found from 200 runs of the CMA-ES algorithm. Fitted to the *14_7* dataset.



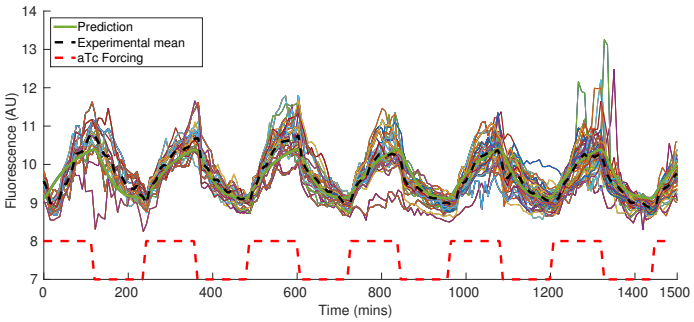
(c) Correlation matrix of estimated parameter sets, from the *13_9* dataset.



(d) Correlation matrix of estimated parameter sets, from the *14_7* dataset.

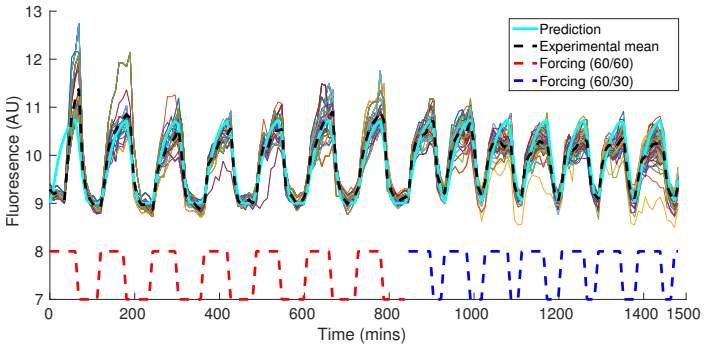


(e) Model prediction, using the parameter set with the smallest error value of the initial 200 found, for the *13_9* dataset.

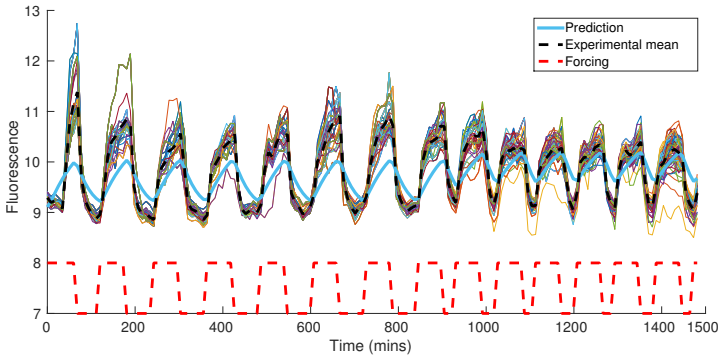


(f) Model prediction, using the parameter set with the smallest error value of the initial 200 found, for the *14_7* dataset.

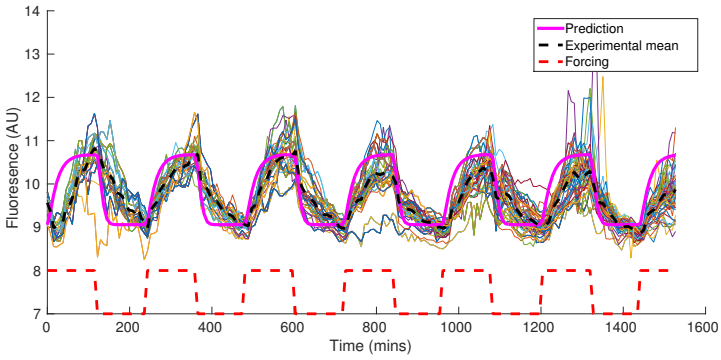
Fig. 6: Parameter estimates, inter-parameter correlation values, and model predictions for the *13_9* and *14_7* datasets. Note that in the model predictions, aTc forcing is shown - IPTG concentration is constant at a level which saturates the cell's response. The forcing curve's height is schematic - aTc concentration is switched between off, and a level which saturates the cell's response.



(a) Model trained on *13_9* 60/60 data only, full prediction



(b) *14_7* prediction *13_9* data



(c) *13_9* prediction *14_7* data

Fig. 7: Cross validating data by taking parameter estimates from one dataset, and using them to predict another. 7a shows model predictions for the full *13_9* dataset, when only trained on the first forcing period data. 7b, 7c show model predictions for fitting to one dataset, and predicting the other.

B. Parameter Estimability

There are two main reasons why a parameter may not be estimable [16]–[18]: Model predictions may be insensitive to the value of a particular parameter, or the effects of varying one parameter on model predictions may be highly correlated with the effects of varying several others.

These problems may stem from structural inadequacies in the model (often termed *structural identifiability*), in which two different parameter sets can give identical model predictions [19], [20]. If this is the case, no amount of experimental data will allow us to estimate parameters, and we must consider reformulating the model. [6]

Problems may also arise for more practical reasons (*practical identifiability* [16]). For example, it is possible that in the experimental regime we operate in, parameter effects may be weak, or highly correlated, but in other regimes this is not true. [7] In this case, parameter estimates may be improved by taking data in more varied experimental conditions, and attempting to observe as many components of the model output as possible.

We may begin to investigate these issues in our model by performing a local sensitivity analysis about one of the solutions found in our initial parameter estimation. We numerically estimate the sensitivity matrix, S:

$$S_{ij} = \hat{\theta}_j \frac{\partial z}{\partial \theta_j}\Big|_{t_i}, \tag{11}$$

where S_{ij} is the derivative of the observed fluorescence, z, with parameter θ_j, evaluated at timepoint t_i. Each column of S is a time series of sensitivity co-efficients, which describe how sensitive z is to perturbations in the parameter associated with that column, and at what times it is most sensitive. $\hat{\theta}_j$ is the value of the parameter that the derivative is being evaluated at. It is included to set the scale that parameters may vary at, to ensure that apparently small sensitivity values do not result from a poor choice of units.

It can be shown that if the sensitivity curves (columns of S) are linearly dependent over the range of observation values, the associated parameters cannot be simultaneously estimated [18]. Related to this fact, a number of measures have been proposed to assess parameter estimability from the sensitivity matrix [16], the simplest of which is to plot the sensitivity curves as a function of time, and visually check for obvious linear relations between them.

Fig. 8 shows plots of each column of S, evaluated at the parameter set giving the lowest error in the *13_9* dataset. Fig. 8a shows them unscaled, Fig. 8b shows them scaled by the norm of each column of S, so that their shapes may be more easily compared.

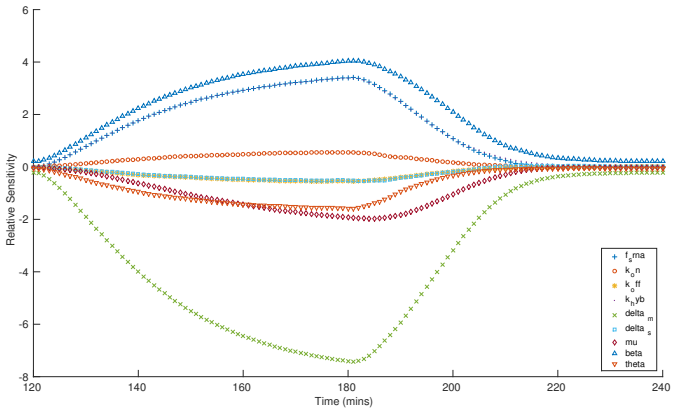
We see that many of the parameters give sensitivity curves of similar shapes, and have near linear dependence - this implies that the effects of perturbing any one of these parameters all look similar in terms of model output, and are hard to distinguish between. This may help to explain why some of our initial parameter estimates are very loose, and why there are correlations between the estimated parameter sets - this result

[6] A simple example of a structurally non - identifiable model is $y = \beta_1\beta_2 x$, where we are given data (x, y), and asked to estimate parameters β_1 and β_2 - we can see that, in principle, only the product $\beta_1\beta_2$ can ever be estimated, a problem no amount of data can fix.

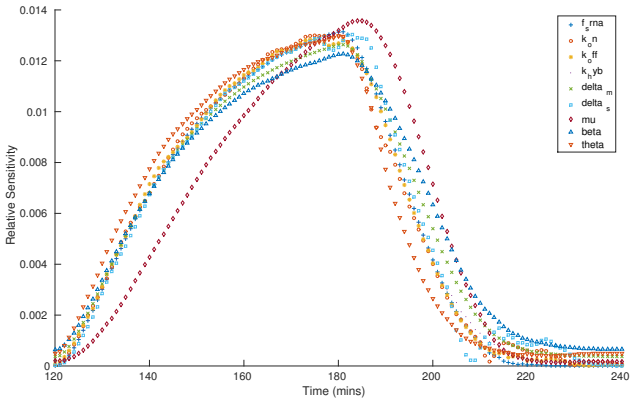
[7] [18] gives an example in which a parameter only affects model predictions after several hours, though others will affect it at all timescales - in this example, if we only took data for a few minutes, the parameter would be inestimable, but in principle it is not.

$$m = \frac{1}{2k_{\text{on}}\left(\mu+\delta_m\right)\left(k_{\text{hyb}}+\delta_m\right)}\left(\sqrt{2(\text{am}+\text{as})k_{\text{on}}\left(\mu+\delta_m\right)\left(\mu+\delta_s\right)\left(k_{\text{hyb}}+\delta_m\right)\left(k_{\text{hyb}}+k_{\text{off}}+\delta_m\right)+(\text{am}-\text{as})^2k_{\text{on}}^2\left(k_{\text{hyb}}+\delta_m\right)^2+\left(\mu+\delta_m\right)^2\left(\mu+\delta_s\right)^2\left(k_{\text{hyb}}+k_{\text{off}}+\delta_m\right)^2}\right. \tag{12}$$
$$\left.-(\text{am}-\text{as})k_{\text{on}}\left(k_{\text{hyb}}+\delta_m\right)+\left(\mu+\delta_m\right)\left(\mu+\delta_s\right)\left(k_{\text{hyb}}+k_{\text{off}}+\delta_m\right)\right)$$

$$c = \frac{k_{\text{hyb}}}{2k_{\text{on}}\left(\mu+\delta_m\right)\left(k_{\text{hyb}}+\delta_m\right)^2}\left(\sqrt{2(\text{am}+\text{as})k_{\text{on}}\left(\mu+\delta_m\right)\left(\mu+\delta_s\right)\left(k_{\text{hyb}}+\delta_m\right)\left(k_{\text{hyb}}+k_{\text{off}}+\delta_m\right)+(\text{am}-\text{as})^2k_{\text{on}}^2\left(k_{\text{hyb}}+\delta_m\right)^2+\left(\mu+\delta_m\right)^2\left(\mu+\delta_s\right)^2\left(k_{\text{hyb}}+k_{\text{off}}+\delta_m\right)^2}\right. \tag{13}$$
$$\left.+(\text{am}+\text{as})k_{\text{on}}\left(k_{\text{hyb}}+\delta_m\right)+\left(\mu+\delta_m\right)\left(\mu+\delta_s\right)\left(k_{\text{hyb}}+k_{\text{off}}+\delta_m\right)\right)$$



(a) Unscaled



(b) Scaled

Fig. 8: Sensitivity coefficients S_{ij} evaluated about a set of estimated parameters from the *13_9* dataset, for a single oscillation. 8a shows them unscaled, 8b shows them scaled by the norm of each column of S.

suggests there is a family of parameters all of which, in terms of the model output we have available, cannot be resolved. As such, we should view estimates of these parameters with extreme caution. Note that the sensitivity curve that looks least similar to the others in Fig. 8b - μ - corresponds to a relatively tightly estimated value in Fig. 6.

C. Differing timescales within the system

Fig. 9 shows model output for all state variables, using the parameter values giving the lowest error on the *13_9* dataset, and normalised to lie on the same scale. We see that $s, m, s:m$ and c respond rapidly to the forcing function, flipping between the two fixed points defined by the step function forcing almost instantly. By contrast, there is delay in the response of p and g, on a timescale comparable to the period of the forcing. This result suggests that, at least in some of the parameter sets initially estimated, the system may have two timescales in it - a fast timescale in which (1) - (4), representing the hybridization of the sRNA and mRNA into a stable complex, equilibrate in response to external forcing, and a slower timescale, in (5), (6), in which measured fluorescence changes in response to the forcing.

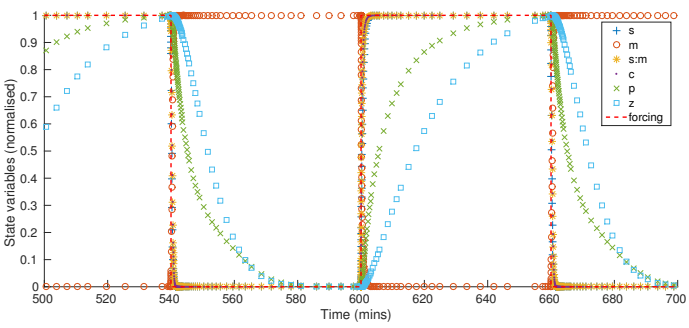


Fig. 9: Model output for all state variables over two oscillations, using the parameter values giving the lowest error on the *13_9* dataset, and normalised to lie on the same scale. Note g is not explicitly shown, but is simply a rescaling of z and as such would lie over it.

If this is the case, then it may be that our experimental data can only probe the system via the fixed point of (1) - (4) , and that the parameters contained within those equations can only work to set the particular fixed point values of p and g that the system tends toward. This would explain the similarity between the sensitivity curves of many parameters - if the parameters in (1) - (4) can only act to alter the model's fixed point, they would all alter the sensitivity curve in exactly the same way, since a small perturbation in one of these parameters would only act to perturb the location of the models fixed

points. It would also suggest that all parameter sets giving the same value of the models fixed point will give similar model predictions - and there may be many, very different, parameter sets which all give the same fixed point.

The fixed point values of $s, m, s : m$ and c affect (5), (6) through the $\beta m + f_s \beta c$ term in (5). Explicit forms for m and c are given in (12), (13). The fact that this term is the only one in (5), (6) to include f_s and β may also explain the similarity of the sensitivity curves for these parameters to those found in (1) - (4) (Fig. 8b) - all that p and g see is this $\beta m + f_s \beta c$ flipping between its two fixed point values, so any parameters that only enter the system via this term will have indistinguishable roles.

Although we have seen that parameter sets with similar least squares error values can contain very different parameter values, this explanation implies that we would expect similar values of $\beta m + f_s \beta c$ across all the sets. Fig. 10 shows a scatterplot of $\beta m + f_s \beta c$ values against error function value, and demonstrates that, though individual parameter values can be spread across very large ranges, they are correlated in such a way as to give similar model fixed points.

In Fig. 11, we also see a strong positive correlation between the fixed point values and the values of μ ($R = 0.9854$ for the visually tightly clustered data). Taken together with the relatively tightly constrained values of μ found in Fig. 6, these results suggest that in order to minimize error, the algorithm is effectively trying to vary $\beta m + f_s \beta c$ and μ, and finding a 'trench' of highly correlated values, which is shown in Fig. 11.

These results further suggest that, while we are trying to minimise model error over our initial high dimensional space, we are effectively working in a lower dimensional space, in which one dimension is the value of the models $\beta m + f_s \beta c$.

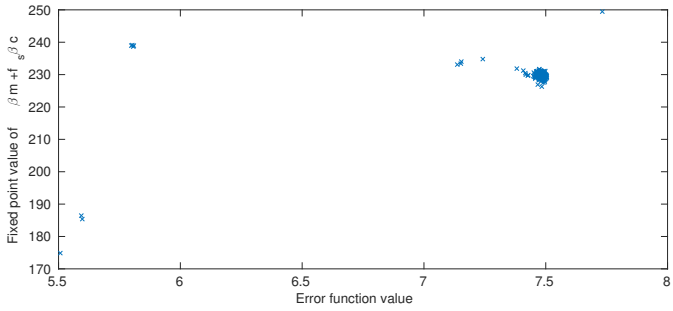


Fig. 10: A scatterplot of the fixed point value of $\beta m + f_s \beta c$, for a given parameter set, against the error value of that set.

IV. Conclusions and Further work

In this report, we have presented a system of ODE's to model a recently proposed synthetic RNA regulatory circuit [4]. We have attempted to estimate the model's parameters using existing time series data, with a least squares minimisation approach, similar to that found in recent system biology literature [13]. Where this approach has failed to give estimates, we have discussed in detail how the method has broken down.

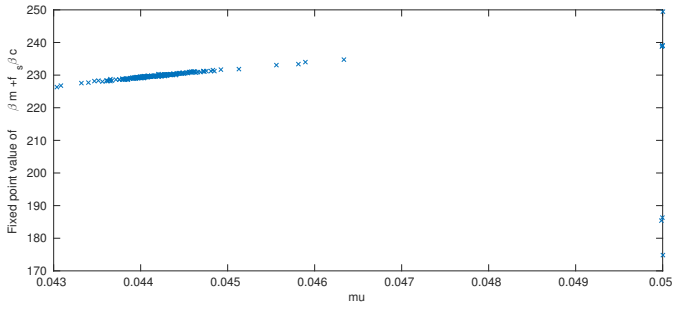


Fig. 11: A scatterplot of the fixed point value of $\beta m + f_s \beta c$, for a given parameter set, against the value of μ in that set.

The results in this paper suggest that, even though the model has a solid biological rationale, some parameters included in it may not be estimable. We have suggested practical reasons for this, namely that the effects of the parameters are highly correlated with one another, because the relevant quantity for determining a model prediction is not the full parameter set, but only the fixed point values of $\beta m + f_s \beta c$, and μ. Thus the parameters in (1) - (4) only work to alter the fixed point value of $\beta m + f_s \beta c$, and so all have a qualitatively identical effects on model predictions. Effectively our algorithm is only working in a 2 dimensional space, varying $\beta m + f_s \beta c$ and μ to minimize error. This result suggests that using any additional step function data available may not help in determining the parameters - all of these step functions are applied at timescales over which the 'fast' part of the model responds instantly, and so essentially tell us the same thing about the parameters contained in those equations. The different sets may give more information about μ, which appears to work on a timescale comparable to the forcing.

We note that it is possible that this separation of timescales is an artefact of the fitting, and not necessarily true of the system. We can only evaluate model predictions on the parameter sets the algorithm returns us, and it seems the case that the parameter sets found by the algorithm have this feature. Hypothetically, there may exist parameter sets where all state variables vary on a similar timescale - all we can say is that, in the search region specified, the algorithm has not found any which give a low error.

This point raises another issue - the arbitrary nature of the initial bounding box. Without good biological constraints on most parameter values, they have been bounded loosely and not necessarily sensibly. Rate constants can vary by many orders of magnitude, and it may be possible that the true optimum lies entirely outside the bounding box specified. Equally, when our fitness function has turned out to be very flat in some directions, or multimodal, good biologically motivated parameter bounds may be needed to ensure our minimisation algorithm does not wander outside realistic parameter space - the algorithm may present several equally good regions of parameter space, and improved bounds will help us choose between them. We also note that while some parameters may be hard to get any realistic bounds on, values for others - μ, Θ

for example - may already be available.

The model may also have general structural issues. [16], [20] give general tests which can be applied to a system of ODE's to determine if all parameters are, in principle, estimable. We note that our system has few enough parameters to make these tests feasible.

Thus, further work may consist of tightening the bounds on parameter values, carrying out a structural analysis, and possibly simplifying the model. A much simpler phenomenological model, consisting of just (5) and (6), may explain existing data perfectly well without the need for inestimable parameters. Experimentally, one option may be to use different forcing functions. The effect of this on parameter estimability might be anticipated by running a sensitivity analysis using this new forcing, to see if the form of Fig. 8 (which shows model predictions for all states) changes. Another option, though perhaps experimentally infeasible, would be to directly observe other components of the state vector - s, m, $s:m$ and c. This would improve parameter estimates by giving direct observations of the effects of parameters contained in (1) - (4).

Some earlier fitting work has been done on this dataset via a similar method to the one used above. Our results suggest that those estimates should be viewed with caution - the estimates will not be unique, and the values of some parameters (those in (1) - (4)) may be very far off any values which are eventually found.

These results also suggest general problems with our methodology - using a least squares minimisation approach. The first problem is local minima. Using the CMA-ES will give better results than a local minimisation algorithm, but you can never be sure you have found a global optimum - for example, [12] compares several algorithms and finds, in the problem set, none locate the true global optimum. A related problem is that of working with single points rather than distributions on parameter space. The CMA-ES will only ever find a single optimum, but in the case where the error landscape is very flat - for example a flat basin pocketed by many local minima - the exact minimum it finds paints a misleading picture, because it gives no idea of the uncertainty in the answer given. To surmount this problem, we may rerun the algorithm many times, from many starting locations, and interpret the resulting spread of parameter values found as an uncertainty [13], as we have done in this report. However this methodology is rather ad hoc.

A more systematic approach might be to use Markov Chain Monte Carlo (MCMC) for the parameter estimation [21], [22]. This method would explicitly provide us with marginal distributions of parameters, giving a more complete and systematic picture of the best parameter sets and their uncertainties than we currently have - we would still be able to pick out a Maximum Likelihood point estimator, as we do now, but rather than getting an ad hoc picture of the uncertainty in the parameters by repeatedly running the minimisation algorithm, the marginals would give us this information directly.

write section on GP's

.

REFERENCES

[1] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter., *Molecular Biology of the Cell*, 4th ed. Garland Science, 2002.

[2] F. J. Isaacs, D. J. Dwyer, and J. J. Collins, "RNA synthetic biology." *Nature biotechnology*, vol. 24, no. 5, pp. 545–554, 2006.

[3] G. Rodrigo, T. E. Landrain, S. Shen, and A. Jaramillo, "A new frontier in synthetic biology: Automated design of small RNA devices in bacteria," pp. 529–536, 2013.

[4] G. Rodrigo, T. E. Landrain, and A. Jaramillo, "De novo automated design of small RNA circuits for engineering synthetic riboregulation in living cells," *Proceedings of the National Academy of Sciences*, vol. 109, no. 38, pp. 15 271–15 276, 2012.

[5] T. Soper, P. Mandin, N. Majdalani, S. Gottesman, and S. a. Woodson, "Positive regulation by small RNAs and the role of Hfq." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 107, no. 21, pp. 9602–9607, 2010.

[6] J. Shine and L. Dalgarno, "Identical 3'-terminal octanucleotide sequence in 18S ribosomal ribonucleic acid from different eukaryotes. A proposed role for this sequence in the recognition of terminator codons." *The Biochemical journal*, vol. 141, no. 3, pp. 609–615, 1974.

[7] R. Lutz and H. Bujard, "Independent and tight regulation of transcriptional units in escherichia coli via the LacR/O, the TetR/O and AraC/I1-I2 regulatory elements," *Nucleic Acids Research*, vol. 25, no. 6, pp. 1203–1210, 1997.

[8] A. Jaramillo, "Predicitve Modelling of Riboregulatory Circuits to Re-engineer Living Cells." [Online]. Available: http://www2.warwick.ac.uk/fac/sci/wcpm/seminars/wcpm_seminar_presentation_alfonso_jaramillo.pdf

[9] Uri Alon, *An Introduction to Systems Biology*, 1st ed.

[10] G. L. Hersch, T. a. Baker, and R. T. Sauer, "SspB delivery of substrates for ClpXP proteolysis probed by the design of improved degradation tags." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 33, pp. 12 136–12 141, 2004.

[11] D. Brewer, M. Barenco, R. Callard, M. Hubank, and J. Stark, "Fitting ordinary differential equations to short time course data." *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, vol. 366, no. 1865, pp. 519–544, 2008.

[12] E. Algorithms, E. Algorithms, C. G. Moles, C. G. Moles, P. Mendes, P. Mendes, J. R. Banga, and J. R. Banga, "Parameter Estimation in Biochemical Pathways: A Comparison of Global Optimization Methods," *Genome Research*, pp. 2467–2474, 2003.

[13] C. Y. Hu, J. Varner, and J. B. Lucks, "Generating effective models and parameters for RNA genetic circuits," *ACS Synthetic Biology*, p. 150605124221004, 2015. [Online]. Available: http://pubs.acs.org/doi/abs/10.1021/acssynbio.5b00077

[14] N. Hansen, "The CMA evolution strategy: A comparing review," *Studies in Fuzziness and Soft Computing*, vol. 192, no. 2006, pp. 75–102, 2006.

[15] ——, "The CMA evolution strategy: A tutorial," *Vu le*, pp. 1–34, 2011. [Online]. Available: http://www.lri.fr/~hansen/cmatutorial110628.pdf

[16] K. a. P. Mclean and K. B. Mcauley, "Mathematical modelling of chemical processes-obtaining the best model predictions and parameter estimates using identifiability and estimability procedures," *Canadian Journal of Chemical Engineering*, vol. 90, no. 2, pp. 351–366, 2012.

[17] K. Z. Yao, B. M. Shaw, B. Kou, K. B. McAuley, and D. W. Bacon, "Modeling Ethylene/Butene Copolymerization with Multisite Catalysts: Parameter Estimability and Experimental Design," *Polymer Reaction Engineering*, vol. 11, no. 3, pp. 563–588, 2003.

[18] J. Beck, *Parameter Estimation in Engineering and Science*, 1st ed. Wiley, 1977.

[19] J. E. Jiménez-Hornero, I. M. Santos-Dueñas, and I. García-García, "Structural identifiability of a model for the acetic acid fermentation process," *Mathematical Biosciences*, vol. 216, no. 2, pp. 154–162, 2008.

[20] M. Grewal and K. Glover, "Identifiability of linear and nonlinear

dynamical systems," *IEEE Transactions on Automatic Control*, vol. 21, no. 6, pp. 833–837, 1976.

[21] J. J. Jitjareonchai, P. M. Reilly, T. a. Duever, and D. B. Chambers, "Parameter Estimation in the Error-in-Variables Models Using the Gibbs Sampler," *Canadian Journal of Chemical Engineering*, vol. 84, no. February, pp. 125–138, 2006.

[22] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning," *Machine Learning*, vol. 50, no. 1-2, pp. 5–43, 2003.

[23] J. B. Andersen, C. Sternberg, L. K. Poulsen, S. P. Bjø rn, M. Givskov, and S. r. Molin, "New unstable variants of green fluorescent protein for studies of transient gene expression in bacteria," *Applied and Environmental Microbiology*, vol. 64, no. 6, pp. 2240–2246, 1998.

[24] R. Iizuka, M. Yamagishi-Shirasaki, and T. Funatsu, "Kinetic study of de novo chromophore maturation of fluorescent proteins," *Analytical Biochemistry*, vol. 414, no. 2, pp. 173–178, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.ab.2011.03.036

APPENDIX A
INITIAL EXPERIMENTAL DATA

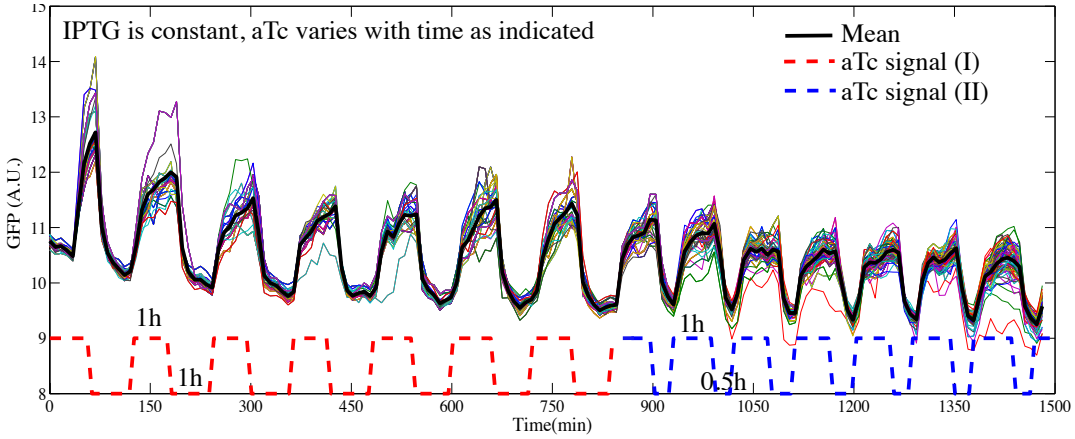


Fig. A.1: The *13_9* dataset, with aTc forcing shown. IPTG concentration is constant at a level which saturates the cell's response. Note the forcing curve's height is schematic - aTc concentation is switched between off, and a level which saturates the cell's response

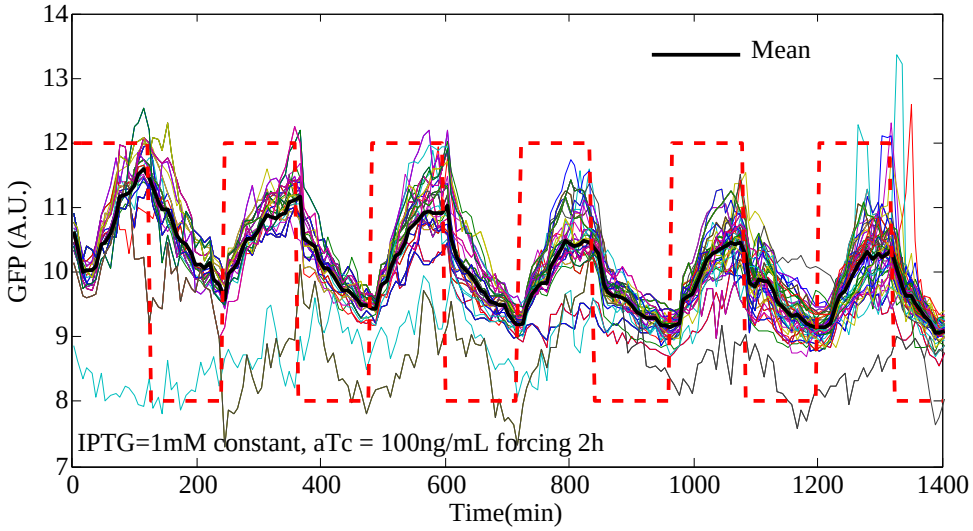


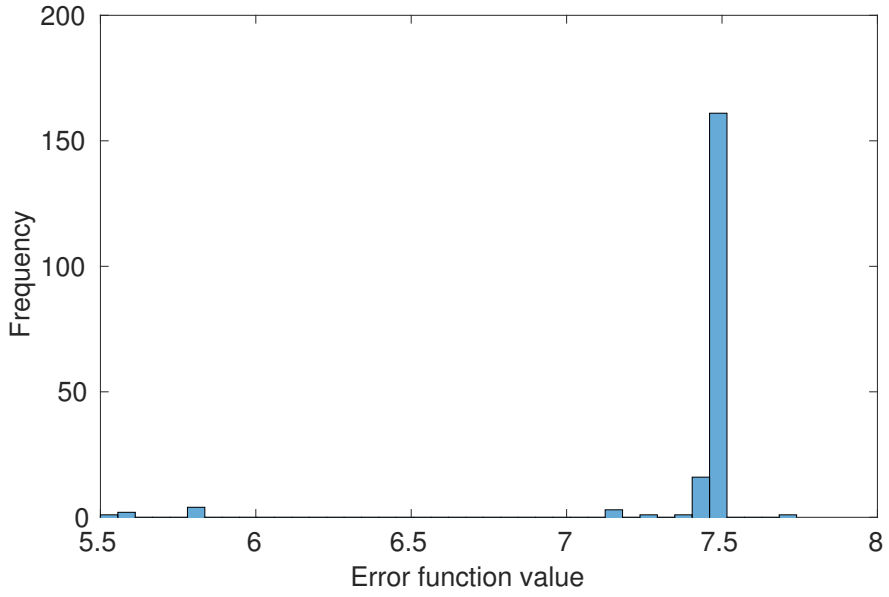
Fig. A.2: The *14_7* dataset, with aTc forcing shown. IPTG concentration is constant at a level which saturates the cell's response. Note the forcing curve's height is schematic - aTc concentation is switched between off, and a level which saturates the cell's response

APPENDIX B
PARAMETER LITERATURE REVIEW

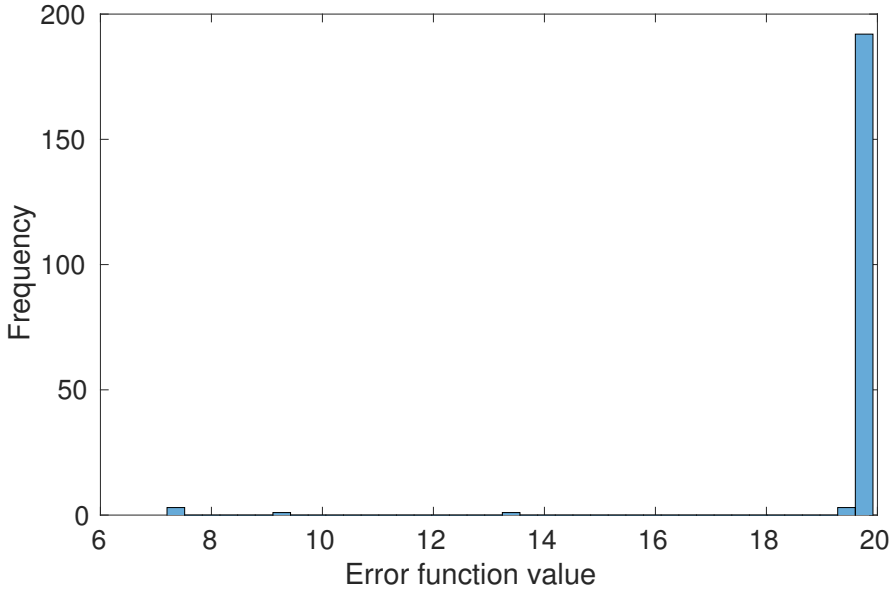
TABLE III: Literature references, or initial rough bounds, on parameter values, with those to be estimated shown in bold

Parameter	Value	Definition	Reference	Initial Bounds
N	300	Number of copies of promoter existing on plasmid DNA	Experimentally set	
z_0	9 AFU	Baseline experimental fluorescence	Experimentally determined	
α_L	11 nM/min	Maximal transcription rate of $P_{\mathrm{LlacO}-1}$ promoter	[7]	
α_T	11 nM/min	Maximal transcription rate of $P_{\mathrm{LtetO}-1}$ promoter	[7]	
f_L	620	Unitless ratio between repressed and unrepressed $P_{\mathrm{LlacO}-1}$ transcription rate	[7]	
f_T	2535	Unitless ratio between repressed and unrepressed $P_{\mathrm{LtetO}-1}$ transcription rate	[7]	
δ_g	0.0005 /min	GFP degradation rate	[23]	
γ	0.132 /min	GFP maturation rate	[24]	
v_z	100 nM/min	degradation constant of clpx	[10]	
K_z	75 nM/min	Dissociation constant of clpx	[10]	
$\boldsymbol{\Theta}$	nM/AFU	Ratio between GFP concentration and observed fluorescence		300 - 1000
$\boldsymbol{\mu}$	/min	Dilution rate		0.001-0.05
$\boldsymbol{\delta_m}$	/min	mRNA degredation rate		1 - 10^5
$\boldsymbol{\delta_s}$	/min	sRNA degredation rate		1 - 10^3
δ_{sm}	/min	Unstable sRNA:mRNA degradation rate		Set to δ_m
δ_c	/min	Stable sRNA:mRNA degradation rate		Set to δ_m
$\boldsymbol{k_{on}}$	/min	sRNA:mRNA binding rate		100 - 10^7
$\boldsymbol{k_{off}}$	/min	sRNA:mRNA unbinding rate		1 -10^8
$\boldsymbol{k_{hyb}}$	/min	sRNA:mRNA hybridization rate		1 - 10^4
$\boldsymbol{\beta}$	/min	Baseline translation rate of repressed mRNA		0.0001 - 10
$\boldsymbol{f_s}$		Ratio of repressed mRNA to unrepressed complex translation rate.		0.1 - 10^4

APPENDIX C
HISTOGRAM OF INITIAL FIT ERROR VALUES



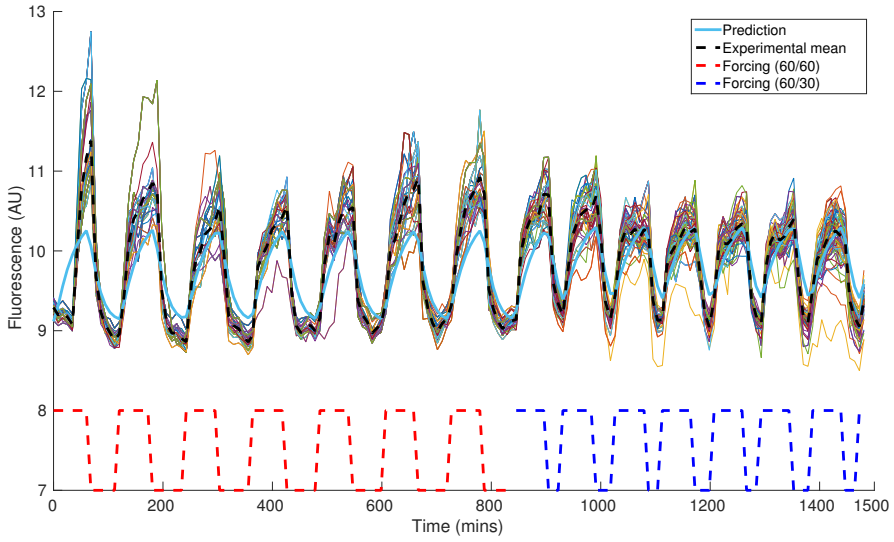
(a) Error values found from 200 initial parameter estimates, *13_9 dataset*.



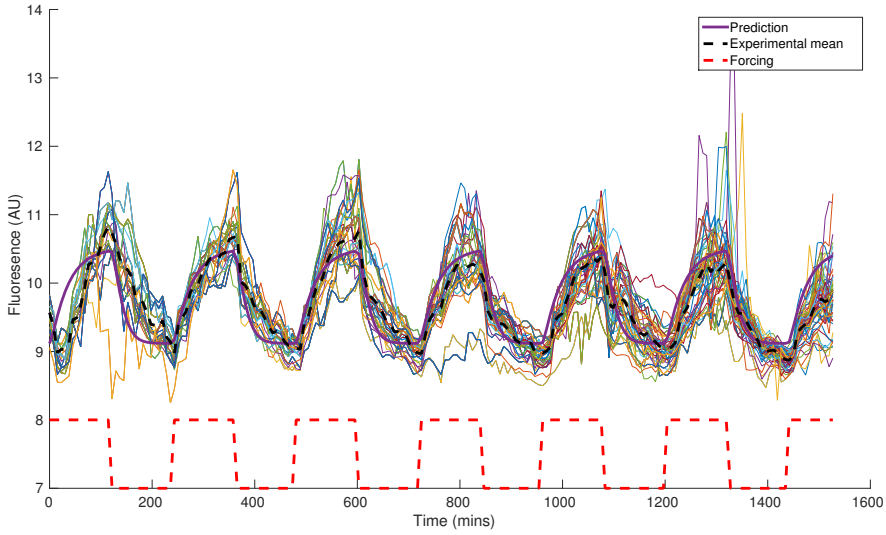
(b) Error values found from 200 initial parameter estimates, *14_7 dataset*

Fig. C.1

APPENDIX D
INITIAL PARAMETER ESTIMATES, BOTH DATASETS



(a) Model prediction from parameter set which gives lowest error when trained on the combination of both datasets, plotted against the *13_9* dataset.



(b) Model prediction from parameter set which gives lowest error when trained on the combination of both datasets, plotted against the *14_7* dataset.

Fig. D.1

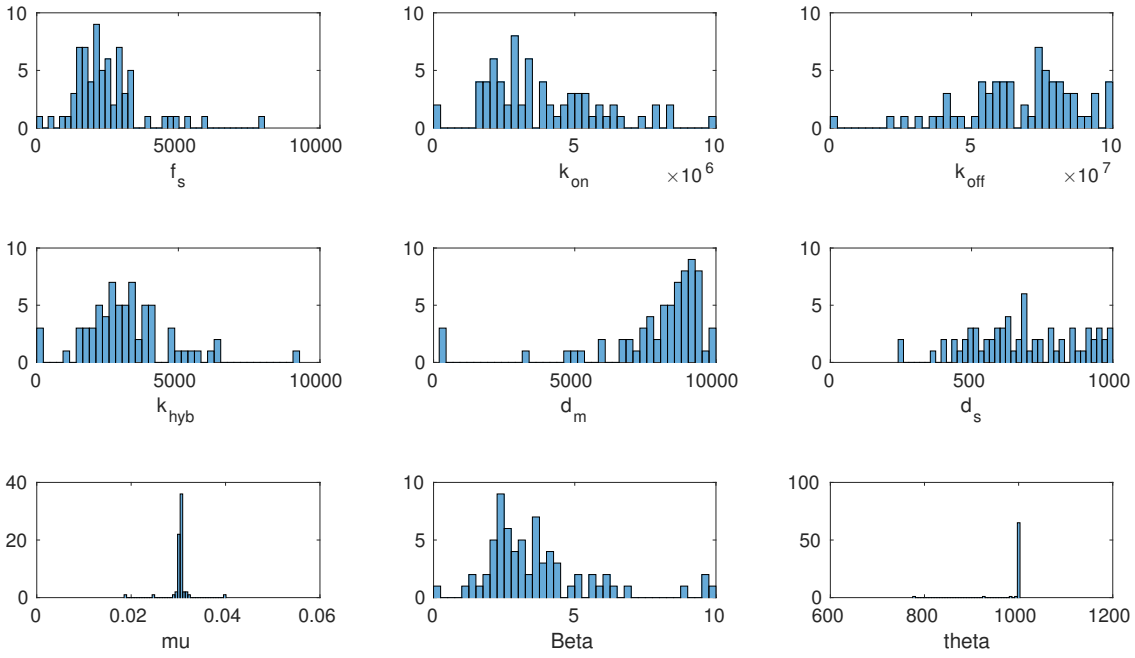


Fig. D.2: Histogram of estimated parameter values, found from 100 runs of the CMA-ES algorithm. Fitted to both the *13_9* and *14_7* datasets.

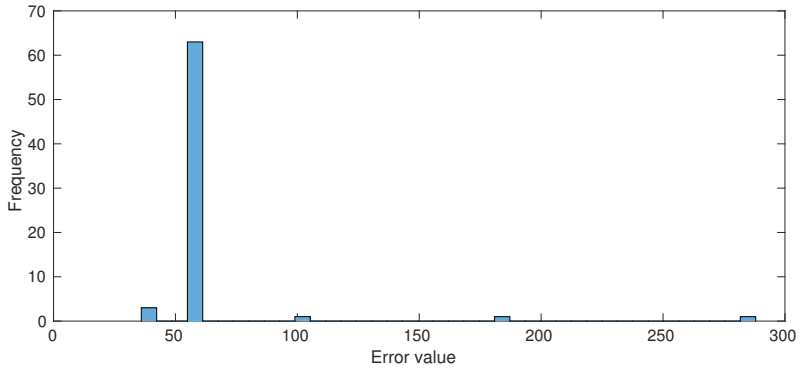


Fig. D.3: Error values found from 100 initial parameter estimates. Fitted to both the *13_9* and *14_7* datasets.