Model-Based Batch Variability Reduction in Despite Parameter Structural Non-Identifiability

Vipul Singhal
Spatial and Single Cell Systems
Genome Institute of Singapore
60 Biopolis St, Singapore, 138672
vipuls@gis.a-star.edu.sg

Richard M. Murray
Control and Dynamical Systems
California Institute of Technology
Pasadena, CA 91125, USA
murray@cds.caltech.edu

Abstract

A system's behavior can be affected by the environment that it operates in, which can lead to batch variability in measurements. If data are to be made independent of the environment they were collected in, such variability must be compensated for. One approach involves performing calibration measurements on individual environments, and using these with parameterized models to predict system behavior in a reference environment, with the effects of environmental variation removed. This calibration-correction workflow can be challenging in biological contexts, because biological models tend to have many more parameters than measurable quantities, such that the outputs of the system are insufficiently informative for identifying parameter values uniquely. In this study, we define a model-based calibration procedure for reducing system variability across environments, and provide a set of parameter consistency conditions under which the prediction steps in the procedure are guaranteed to work, even in the presence of parameter non-identifiability. We demonstrate our results on real and simulated data examples from synthetic biology.

1 Introduction

Batch- or environment-specific effects in the measured behavior of a system can be a major confounding factor in data-intensive workflows. Calibration measurements performed on individual environments may be used, in conjunction with parametrized dynamical models of the systems under consideration, to correct for these variations. Such 'calibration and correction' methodologies [1] involve identifying parameters and transferring them across models in different environments. This task can be complicated by the fact that model parameters may not be identifiable, even in a structural[1] sense.

In this paper, we consider systems comprising *processes* operating in different *environments*, and describe the problem of reducing the variability in the observed behavior of a process across environments in terms of what we call the *data correction problem*. Solving this problem involves finding a method for transforming the observed behavior of a process from a given (*candidate*) environment into what it would have looked like had it been collected in a *reference* environment. The idea being that whenever data are collected in a given environment, they should be transformed into their reference environment versions, making it possible to directly compare measurements from different environments.

Once the data correction problem has been defined, we describe a model-based calibration procedure called the *calibration-correction method* for solving it. This procedure is based on

[1] See Definitions 1 - 3 below.

Preprint. Under review.

a similar method used for correcting wind-tunnel data collected across different test facilities [1]. The assumption underlying the method is that there are certain features that vary from environment to environment, and the variation of these features can be captured as the variation of certain *environment-specific parameters*. Furthermore, we assume the parameters associated with the process (*process-specific parameters*) whose behavior we are interested in do not change across environments. In general, we expect this assumption to hold for sufficiently fine-grained models, becoming more approximate when coarser models are used.

The calibration-correction method involves first performing a set of calibration experiments on the reference and candidate environments, and using corresponding calibration models to estimate the environment specific parameters associated with each environment. Subsequently, the behavior of a *test* process, whose behavior we are interested in transforming across environments, is measured in the candidate environment, and its process specific parameters are estimated from the measured data and a corresponding model. This estimation step is performed with the environment specific parameters for the candidate environment fixed at the values obtained at the calibration stage. Finally, the prediction for the process behavior in the reference environment is generated using the test process model's process specific parameters, along with the environment specific parameters for the reference environment. The data correction problem and calibration-correction method are defined in Section 4 below.

In the description of the calibration-correction method above, we use parameters from different parameter estimation steps in the final test process model. When these parameters are structurally non-identifiable, combining parameters in this manner can fail. In Sections 5-7, we describe a set of necessary and sufficient *consistency conditions* under which the non-identifiability of parameters does not hinder the calibration procedure.

Our results will rely on the fact that the set of structurally non-identifiable parameters is an equivalence class with respect to the observed behavior (i.e., outputs) of a system for a given set of initial conditions and inputs [2], and that the data being transformed involve only output variables, and not the trajectories of the full set of state variables. We also show that these consistency conditions may be violated when the non-identifiability possesses a certain type of parametric covariation, and describe a modification to the methodology that allows it to meet the consistency conditions in the presence of such covariation (Section 6). Throughout this study, we demonstrate the definitions and results on an example involving real experimental data, and demonstrate some of the more theoretical results using simulated data (Section 8).

2 Motivating Case Study: Reduction of Variability in Genetic Circuit Behavior across Cell Extracts

In this section, we describe an example from synthetic biology that motivates the development of the calibration framework and the identifiability results described in this study. A primer on the key biological terminology used here can be found in Appendix E.

2.1 Circuits and Extracts

A key task in synthetic biology is the design of genetic *circuits* [3, 4], which are networks of interacting genes, proteins, and other biological molecules. The simplest examples of circuits are the expression of a protein under a constitutively (constantly) expressed promoter, or the repression of a protein's expression by a transcription factor protein, which is itself expressed constitutively. More complex circuits include the incoherent feed-forward loop [5], which generates pulses by first expressing a protein, and then repressing it after a time delay, or the so-called 'repressilator' [3], which comprises a network of three proteins repressing one another in a cyclical fashion and displaying oscillatory behavior due to the time delays involved in protein expression.

Such circuits may be used in various applications, such as the production of pharmaceuticals and medical therapeutics [6], drug discovery [7] and biosensors [8]. The design of these circuits in live cells can be time consuming [9], both because of the difficulty of the process of cloning the genetic circuits onto DNA plasmids, and of incorporating these plasmids into live cells. This, in turn, means that the design-build-test iterations for these circuits can be slow, and has motivated the development of cell extracts as prototyping platforms for iterating over circuit designs, in analogy to electrical circuit breadboards used in electrical engineering or wind tunnels used in aeronautics.

Cell extracts, as their name suggests, are extractions of the cytoplasmic contents of (typically) bacterial or yeast cells. These contents include the molecular machinery needed for the processes of transcription and translation. Once an extract batch has been produced, DNA and raw materials such as nucleotides and amino acids can simply be added to it, and the genetic circuit encoded by the DNA is expressed, displaying measurable dynamics in the concentrations of mRNA and proteins.

2.2 Circuit Variability and Data Correction

Despite being simplified prototyping platforms for genetic circuit design, extracts can display large variability between prepared batches, which limits our ability to reliably generalize experimental data measured in any one extract [10, 11, 12].

Figure 1 shows the expression of six fluorescent proteins under the control of constitutively expressed promoters in each of three extracts. It shows that the three extracts express the same set of circuits at different levels. In the context of the process-environment duality described in the Introduction, each circuit is the process, and the extracts are the environments.

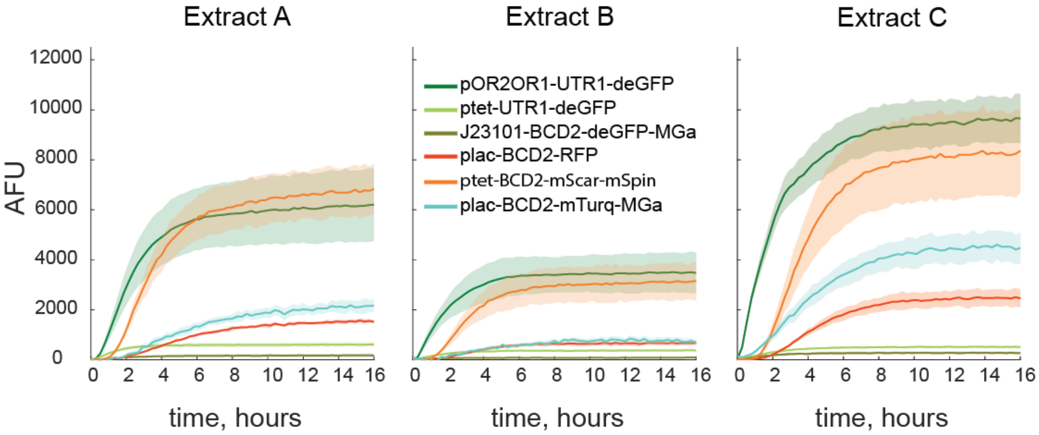

Figure 1: There is significant batch-to-batch variability in extracts. We expressed six constitutive reporter constructs (n = 5, technical repeats, shaded region = standard deviation) in three extract batches prepared by different scientists. Each of the constructs was expressed on linear DNA.

3 Notation and Preliminary Ideas

3.1 Experiments, Systems, Models and Parameters

We consider systems $\mathcal{S} = (\mathcal{E}, \mathcal{P})$ described as a combination of an environment $\mathcal{E}$ and a process $\mathcal{P}$, and define an experiment $\mathcal{H} = (\mathcal{S}, x_0, \overline{y})$ to be the execution of a system under initial conditions x_0 and output measurements $\overline{y}$, where the bar denotes the assumption that experimental data reflect the ground truth. Time dependent inputs may be included without significant change to the results derived in this paper, and are suppressed for simplicity.

The parameter vector θ of a model M associated with a given experiment will be partitioned into environment specific parameter coordinates $\theta_e \in \mathbb{R}^{q_e}$, and process specific parameter coordinates $\theta_p \in \mathbb{R}^{q_p}$, denoted $\theta = (\theta_e, \theta_p)$. We do not restrict these parameters to be in the positive orthant, since any positive parameters may be log-transformed, as we do in the running example throughout this study.

The choice of which parameters are primarily environment specific and which ones are process specific is part of the modeling process, and can require modeler intuition, application domain expertise and some iterative testing. However, a general guideline for defining the partition may still be prescribed: environment specific parameters are parameters associated primarily with components that are present in the system regardless of the the process implemented, while process specific parameters are parameters associated with components that may no longer be present in the system when the process is changed. In the biological example discussed in this study, environment specific parameters are the concentration of transcriptional and translational machinery in the extracts and the elongation rates for transcription and translation (Examples 1–4). Examples of process specific parameters include promoter-transcription factor binding parameters or transcription factor dimerization parameters.

Experiments are modeled using initialized parametrized models with the ordinary differential equations (ODEs) of the form

$$\begin{aligned} \dot{x}(t) &= f(x(t), \theta), \\ y(t, \theta, x_0(\theta)) &= h(x(t), \theta), \qquad x(0) = x_0(\theta). \end{aligned} \tag{1}$$

The state and initial condition vectors are denoted $x(t), x_0(\theta) \in \mathbb{R}^n_+$, where we often suppress t and θ to unclutter notation. The solutions are assumed to exist for all $t \geq 0$, the parameter vector symbol is $\theta = (\theta_e, \theta_p) \in \Omega \subseteq \mathbb{R}^{q_E + q_P}$, where Ω is the set of all parameter values of interest. The output at time t is denoted $y(t, \theta, x_0) \in \mathbb{R}^r$, with $y(\theta, x_0)$ denoting an entire trajectory over the appropriate interval of existence. The functions f and h are assumed to be analytic vector fields with respect to x in some neighborhood of any attainable x, and time dependence of the vector fields can be modeled by including t in the state variables [13], if necessary. Without loss of generality, we do not explicitly model inputs to the system, finite intervals of existence of solutions to our ODEs, or restrictions of the state and parameter spaces to sets smaller than the non-negative orthant. The mathematical framework and arguments we develop do not depend on these simplifications, and the general case can be included if needed. We will use the shorthand $y(\theta, x_0) = M(\theta, x_0)$ to refer to the model in equation (1), and will often suppress arguments such as x_0 for brevity. We will sometimes replace θ with $\left(\theta_e, \theta_p\right)$, as in stating $y\left(\theta_e, \theta_p\right) = M\left(\theta_e, \theta_p\right)$ or just $M\left(\theta_e, \theta_p\right)$ (dropping the inner parentheses), or even $M\left(\left(\theta_e, \theta_p\right), x_0\right)$. We will use the hat symbol (ˆ) to denote an estimated parameter value ($\hat{\theta}$, for instance), or a simulated model trajectory, $\hat{y}$. The tilde (˜) over parameter symbols is reserved for miscellaneous purposes, particularly in proofs.

Example 1 (Enzymatic Reaction ODE Model). We begin by describing the ODE model for the calibration circuit we will use in the running example involving the correction of genetic circuit behavior across extracts. Suppose we have multiple extracts, and consider the constitutive expression of green fluorescent protein (GFP) in the i-th extract. We model this using an enzymatic reaction model, which abstracts transcription and translation into a single step,

$$\mathrm{D_G} + \mathrm{Enz} \underset{k_{\mathrm{rG}}}{\overset{k_{\mathrm{fG}}}{\rightleftharpoons}} \mathrm{D_G{:}Enz} \xrightarrow{k_{\mathrm{c},i}} \mathrm{D_G} + \mathrm{Enz} + \mathrm{G}, \tag{2}$$

where $\mathrm{D_G}$ is the GFP DNA, Enz is an enzyme species denoting a lumped description of the machinery that implements the conversion of DNA into protein, $\mathrm{D_G}$:Enz denotes a bound complex of DNA and enzyme, G is the GFP protein. The rate of production of GFP from the bound complex, $k_{\mathrm{c},i}$, is assumed to be dependent on the extract (having an i subscript), as is the initial enzyme concentration $\mathrm{Enz}_{0,i}$, so that the environment specific parameters are $\theta_{e,i,\mathrm{cal}} = (\mathrm{Enz}_{0,i}, k_{\mathrm{c},i})$. The process specific parameters are $\theta_{p,\mathrm{cal}} = (k_{\mathrm{fG}}, k_{\mathrm{rG}})$, and are assumed to not depend on the extract.

Let the circuit above be implemented at five different initial concentrations of the GFP DNA, with the j-th element denoted $\mathrm{D_{G0,j}} \in \begin{pmatrix} 1 & 2 & 5 & 10 & 20 \end{pmatrix}$ nM. The chemical reaction system has two conservation laws,

$$\begin{aligned} \mathrm{D_{G0,j}} &= [\mathrm{D_G}] + [\mathrm{D_G{:}Enz}], \\ \mathrm{Enz}_{0,i} &= [\mathrm{Enz}] + [\mathrm{D_G{:}Enz}], \end{aligned}$$

$$,$$

which we use to eliminate [Enz] and $\left[\mathrm{D_G{:}Enz}\right]$ from the state space ODE model. The resulting model $y_{i,\mathrm{cal},j} = M_{\mathrm{cal}}\left((\theta_{e,i,\mathrm{cal}}, \theta_{p,\mathrm{cal}}), x_{0,j}\right)$ for the i-th extract and j-th initial DNA concentration (assuming mass-action kinetics) is given by

$$\begin{aligned} \frac{\mathrm{d}[\mathrm{D_G}]}{\mathrm{d}t} &= -k_{\mathrm{fG}}\left(\mathrm{Enz}_{0,i} - \mathrm{D_{G0,j}} + [\mathrm{D_G}]\right)[\mathrm{D_G}] + (k_{\mathrm{rG}} + k_{\mathrm{c},i})\left(\mathrm{D_{G0,j}} - [\mathrm{D_G}]\right), \\ \frac{\mathrm{d}[\mathrm{G}]}{\mathrm{d}t} &= k_{\mathrm{c},i}\left(\mathrm{D_{G0,j}} - [\mathrm{D_G}]\right), \end{aligned} \tag{3}$$

where $[\cdot]$ denotes concentration of a chemical species, and the (i, j)-th model's initial condition is

$$x_{0,j} = \begin{pmatrix} [\mathrm{D_G}] \\ [\mathrm{G}] \end{pmatrix}(0) = \begin{pmatrix} \mathrm{D_{G0,j}} \\ 0 \end{pmatrix}. \tag{4}$$

Note that the environment specific parameter $\mathrm{Enz}_{0,i}$ does not appear in the initial condition vector because we eliminated [Enz] as a state variable from the model. Had we kept it, the initial conditions would have depended on $\mathrm{Enz}_{0,i}$, which could be emphasized by writing $x_{0,j}(\theta_{e,i,\mathrm{cal}}, \theta_{p,\mathrm{cal}})$, or even just $x_{0,j}(\theta_i)$ for simplicity.

The model output trajectories are denoted $y_{i,\mathrm{cal},j}$, the data trajectories $\overline{y}_{i,\mathrm{cal},j}$, and the experiment $\mathcal{H}_{i,j} = (\mathcal{S}_i, x_{0,i,j}, \overline{y}_{i,\mathrm{cal},j})$. ◇

Having multiple experimental conditions for the same underlying system or model is a common occurrence. Modeling data from multiple experimental conditions can be achieved by collecting individual models, one for each experimental condition, and parametrizing them with a single set of parameters. That is, we constrain corresponding parameters in the individual models to take on identical values, a property that is sometimes referred to as 'hard' parameter sharing, or parameter consensus [14]. This is useful when fitting models to data, where the shared parameters simultaneously fit each individual model to corresponding data. We illustrate this explicitly in Example 3 below, where we describe it in the context of the calibration-correction method.

3.2 Parameter Non-Identifiability

In this subsection, we follow Walter and Lecourtier [13] in defining the notion of parameter non-identifiability

Definition 1. Let $M(\theta_A)$ be a parametrized model, and let $M(\theta_B)$ be a model with the same structure. $M(\theta_A)$ and $M(\theta_B)$ are said to be *output-indistinguishable* if

$$\begin{aligned} &\theta_A,\ \theta_B \in \Omega, \\ &y(\theta_A, x_0) = y(\theta_B, x_0) \quad \forall t \geq 0,\ \forall x_0 \in \mathbb{R}^n_+. \end{aligned} \tag{5}$$

◇

Definition 2. The i^{th} coordinate of θ_A, denoted $\theta_{A,i}$, is *structurally globally identifiable* if for almost any $\theta_A \in \Omega$, equation (5) has a unique solution for $\theta_{B,i}$. ◇

This means that the i^{th} coordinate of the parameter vector being structurally globally identifiable is equivalent to the set of parameter points θ_A in the parameter space that differ in their i^{th} coordinate and still give output indistinguishable trajectories having measure zero. Stated differently, for an structurally globally identifiable coordinate, output indistinguishable trajectories almost always lead to a unique estimate of the coordinate.

Definition 3. The model $M(\theta)$ is called *structurally globally identifiable* if all its parameters θ_i, for $i = 1, 2, \ldots, q_E + q_P$, are structurally globally identifiable. ◇

In the absence of global identifiability, multiple points in the parameter space give rise to the same output behavior. In biological applications, this situation tends to be common due to a limited number of measurements and a large number of state variables. Our main goal is to demonstrate that it is not always necessary to achieve global identifiability for every parameter in order to successfully perform a modeling task (such as transferring parameters for batch variability reduction). To this end, we shall consider models with parameters that are not structurally globally identifiable, and thus allow θ_e and θ_p to exist in sets of output-indistinguishable parameters, denoted by E and P respectively.

4 A Calibration and Correction Methodology for Reducing Process Variability Across Environments

In this section, we frame variability reduction formally in terms of what we call the data correction problem, and then define the calibration-correction method as a means of solving this problem. We illustrate both of these definitions with the example of reducing genetic circuit variability across extract batches.

4.1 Framing Variability Reduction as the Data Correction Problem

Consider two environments, a *reference environment* ($\mathcal{E}_1$), and a *candidate environment* ($\mathcal{E}_2$). Let $\mathcal{H}_{i,\text{cal}}$ (resp. $\mathcal{H}_{i,\text{test}}$) be an experiment performed with a *calibration process* $\mathcal{P}_{\text{cal}}$ (resp. *test process* $\mathcal{P}_{\text{test}}$) in the environment $\mathcal{E}_i$. Assume that we may pick models $M_{\text{cal}}(\theta_{\text{cal},i})$ and $M_{\text{test}}(\theta_{\text{test},i})$ corresponding to $\mathcal{H}_{i,\text{cal}}$ and $\mathcal{H}_{i,\text{test}}$ (respectively), as long as the models are at the same level of detail (see Remark 3 and Example 2).

Definition 4 (The Data Correction Problem)**.** Let $\mathcal{H}_{i,\text{test}} = ((\mathcal{E}_i, \mathcal{P}_{\text{test}}), x_{0,\text{test}}, \overline{y}_{i,\text{test}})$, $i = 1, 2$, be the experiments describing the test circuit in the reference and candidate extracts respectively. Assume that we have the freedom to design and perform calibration experiments $\mathcal{H}_{i,\text{cal}}$, $i = 1, 2$, in both the reference and candidate extracts, and collect the resulting data, $\overline{y}_{1,\text{cal}}$ and $\overline{y}_{2,\text{cal}}$. Solving the *data correction problem* involves finding a method that takes as input the tuple $(M_{\text{cal}}, M_{\text{test}}, \overline{y}_{1,\text{cal}}, \overline{y}_{2,\text{cal}}, \overline{y}_{2,\text{test}})$ and returns a trajectory $\hat{y}_{1,\text{test}}$, such that $\hat{y}_{1,\text{test}} = \overline{y}_{1,\text{test}}$. ◇

Remark 1. In general, the data correction problem will only be solvable in the model universe, where the data will be generated as follows. Let $\overline{\theta}_{e1}$ and $\overline{\theta}_{e2}$ be the environment specific parameters for $\mathcal{E}_1$ and $\mathcal{E}_2$ respectively. Let $\overline{\theta}_{p,\text{cal}}$ and $\overline{\theta}_{p,\text{test}}$ be the process specific parameters for the calibration and test experiments respectively. Then the output data in the model universe are

$$\overline{y}_{i,\text{cal}} \triangleq \overline{M}_{\text{cal}}\left(\left(\overline{\theta}_{e,i}, \overline{\theta}_{p,\text{cal}}\right), x_{0,\text{cal}}\right),$$
$$\overline{y}_{i,\text{test}} \triangleq \overline{M}_{\text{test}}\left(\left(\overline{\theta}_{e,i}, \overline{\theta}_{p,\text{test}}\right), x_{0,\text{test}}\right),$$

for $i = 1, 2$. ◇

Remark 2. With real data, the equality $\hat{y}_{1,\text{test}} = \overline{y}_{1,\text{test}}$ in the definition must be replaced with the approximate equality $\hat{y}_{1,\text{test}} \approx \overline{y}_{1,\text{test}}$, or perhaps merely even a requirement of a decrease in the distance (under some metric d) between the predicted and reference trajectories relative to the distance between the reference and candidate extract trajectories, $d(\overline{y}_{1,\text{test}}, \hat{y}_{1,\text{test}}) < \epsilon d(\overline{y}_{1,\text{test}}, \overline{y}_{2,\text{test}})$ for some user defined parameter $\epsilon \in [0, 1)$. ◇

Remark 3. Two models are at the same level of detail if, whenever some mechanism is a part of both models, it has the same mathematical expressions describing it in each model (see models in Examples 1 and 2). The reason for this requirement is that we will be using values of parameters estimated using one model in the other when we attempt to solve the data correction problem. This also raises the interesting possibility of using models at different levels of detail, as long as one model can be (model order) reduced to the other. Then, it might be possible to estimate the parameters in one model, and transform them appropriately before using them in the other model. This is left as a future direction of investigation. ◇

Example 2 (The Data Correction Problem for Extract Variability Reduction)**.** Recall from Section 2 that extracts are environments, and genetic circuits are processes. In what follows, we will refer to *reference* and *candidate extracts*, and *calibration* and *test circuits*. The data correction problem involves finding a method for predicting the behavior of the test circuit in the reference extract, given measurements of the behavior of the calibration circuit in both extracts, and the test circuit in (only) the candidate extract.

Figure 2 describes the data correction problem schematically using real experimental data from two extracts and two circuits. The calibration circuit (Figure 2A) comprises the expression of green fluorescent protein (GFP) under the control of the unrepressed—and therefore constitutively expressing—pTet promoter. Figure 2C shows the time courses of the expression of GFP from this circuit in both extracts at various initial DNA concentrations (1 nM to 20 nM), which constitute the calibration data. These data indicate that the circuit expresses at a higher level in the candidate extract than in the reference extract. We will use the notation

$\{\overline{y}_{1,\text{cal},1}, \overline{y}_{1,\text{cal},2}, \overline{y}_{1,\text{cal},3}, \overline{y}_{1,\text{cal},4}, \overline{y}_{1,\text{cal},5}\}$ for the 5 trajectories—one for each initial condition—that constitute the reference extract ($\mathcal{E}_1$) calibration data, and similarly $\{\overline{y}_{2,\text{cal},j}\}_{j=1}^{5}$ for candidate extract ($\mathcal{E}_2$) data.

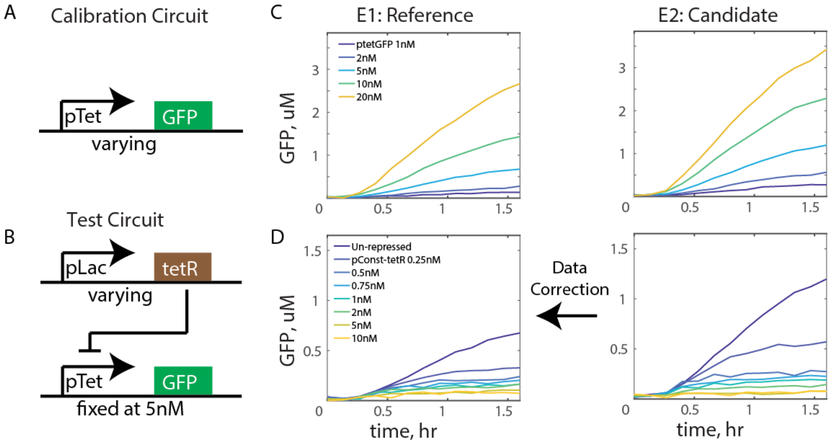

Figure 2: The data correction problem: given a set of (A) calibration and (B) a test circuits (processes), (C) measurements of the calibration circuits in both candidate and reference extracts (environments), models for both the calibration and test circuits, and measurements of the behavior of the test circuit in the candidate extract (D), predict the test circuit's behavior in the reference extract. In (C), the calibration circuit DNA (pTet-GFP) is added at 5 different concentrations ranging from 1 nM to 20 nM, and GFP is measured for 1.6 hours. In (D), the reporter pTet-GFP DNA is fixed at 5 nM, while the repressor protein DNA (pLac-tetR) is varied from 0 to 10 nM (8 total concentrations).

We model this circuit using the enzymatic reaction shown in Example 1, where we define the environment and process specific parameters, initial conditions and model outputs. One reason for picking the one-step enzymatic reaction to model protein production is that, while it can model protein production, it is also simple enough that the joint distribution of its main parameters can be readily visualized in three dimensions. This allows the theoretical conditions we discuss in the next section to be visualized graphically (see Section 8.1 and Figure 7D-F), before being generalized to models with higher dimensional parameter spaces.

The test circuit (Figure 2B) comprises two genes: the tetR repressor gene, whose expression is controlled by a constitutive promoter (pLac), and the GFP reporter gene, which is controlled by the TetR-repressible pTet promoter. Expression of the tetR gene leads to the production of the TetR repressor protein, which decreases how much GFP can be produced by the pTet-GFP DNA. This can be seen in Figure 2D, where (on the left panel, for instance) the individual trajectories show lower expression for higher initial concentrations of the pLac-tetR DNA. We fix the concentration of the repressible pTet-GFP DNA at 5 nM, and vary the concentration of the DNA encoding the repressor protein, pLac-tetR, from 0 to 10 nM. The individual data trajectories are denoted $\{\overline{y}_{1,\text{test},j}\}_{j=1}^{8}$ for the reference extract and $\{\overline{y}_{2,\text{test},j}\}_{j=1}^{8}$ for the candidate extract.

Recall that the models used for calibration and test circuits have to be at the same level of detail (Remark 3) for the data correction problem to be solved by the calibration-correction method. To this end, we use the enzymatic reaction to model protein production once again. We model repression by sequestering free GFP DNA using the (dimerized) TetR protein, resulting in the following chemical equations, The test circuit may be modeled by the equations

$$\begin{aligned}
\mathrm{D_T} + \mathrm{Enz} &\underset{k_{\mathrm{rT}}}{\overset{k_{\mathrm{fT}}}{\rightleftharpoons}} \mathrm{D_T{:}Enz} \xrightarrow{k_{\mathrm{c}}} \mathrm{D_T} + \mathrm{Enz} + \mathrm{T}, \\
\mathrm{D_G} + \mathrm{Enz} &\underset{k_{\mathrm{rG}}}{\overset{k_{\mathrm{fG}}}{\rightleftharpoons}} \mathrm{D_G{:}Enz} \xrightarrow{k_{\mathrm{c}}} \mathrm{D_G} + \mathrm{Enz} + \mathrm{G}, \\
2\mathrm{T} &\underset{k_{\mathrm{r,dim}}}{\overset{k_{\mathrm{f,dim}}}{\rightleftharpoons}} \mathrm{T_2}, \\
\mathrm{D_G} + \mathrm{T_2} &\underset{k_{\mathrm{r,rep}}}{\overset{k_{\mathrm{f,rep}}}{\rightleftharpoons}} \mathrm{D_G{:}T_2},
\end{aligned} \tag{6}$$

where G and T are are the GFP and TetR proteins, $\mathrm{D_T}$ is the DNA that codes for the TetR protein, and $\mathrm{T_2}$ is the TetR protein dimer, which sequesters the GFP expressing DNA ($\mathrm{D_G}$), repressing GFP production. The corresponding ODE model is constructed similarly to that in Example 1. Let $\mathrm{D_{G0}} = 5$ nM be the initial GFP DNA concentration, $\mathrm{Enz}_{i,0}$ the unknown initial enzyme concentration (being one of the environment specific parameters), and $\mathrm{D_{T0,j}}$ the j-th TetR DNA initial condition, as in $\mathrm{D_{T0,j}} \in (0\,,0.25\,,0.5\,,0.75\,,1\,,2\,,5\,,10)$ nM. Note that in the ODE model of this circuit, there are three conservation laws,

$$\begin{aligned}
\mathrm{Enz}_{i,0} &= [\mathrm{Enz}] + [\mathrm{D_T{:}Enz}] + [\mathrm{D_G{:}Enz}] \\
\mathrm{D_{T0,j}} &= [\mathrm{D_T{:}Enz}] + [\mathrm{D_T}] \\
\mathrm{D_{G0}} &= [\mathrm{D_G{:}Enz}] + [\mathrm{D_G}] + [\mathrm{D_G{:}T_2}],
\end{aligned}$$

which can be used to eliminate, say, [Enz], [$\mathrm{D_T}$:Enz] and [$\mathrm{D_G}$:Enz] from the state space ODE model. The resulting model, $\hat{y}_{i,\mathrm{test},j} = M_{\mathrm{test}}\big((\theta_{e,i}, \theta_{p,\mathrm{test}}), x_{0,j}\big)$, is given by

$$\begin{aligned}
\frac{\mathrm{d}[\mathrm{D_T}]}{\mathrm{d}t} &= -k_{\mathrm{fT}}\Big(\mathrm{Enz}_{i,0} - \Big(\mathrm{D_{T0,j}} - [\mathrm{D_T}]\Big) - \Big(\mathrm{D_{G0}} - [\mathrm{D_G}] - [\mathrm{D_G{:}T_2}]\Big)\Big)[\mathrm{D_T}] \\
&\quad + (k_{\mathrm{rT}} + k_{\mathrm{c},i})\Big(\mathrm{D_{T0,j}} - [\mathrm{D_T}]\Big), \\
\frac{\mathrm{d}[\mathrm{D_G}]}{\mathrm{d}t} &= -k_{\mathrm{fG}}[\mathrm{Enz}][\mathrm{D_G}] + (k_{\mathrm{rG}} + k_{\mathrm{c},i})\big(\mathrm{D_{G0}} - [\mathrm{D_G}] - [\mathrm{D_G{:}T_2}]\big) \\
&\quad - k_{\mathrm{f,rep}}[\mathrm{D_G}][\mathrm{T_2}] + k_{\mathrm{r,rep}}[\mathrm{D_G{:}T_2}] \\
\frac{\mathrm{d}[\mathrm{T}]}{\mathrm{d}t} &= k_{\mathrm{c},i}\Big(\mathrm{D_{T0,j}} - [\mathrm{D_T}]\Big) - 2k_{\mathrm{f,dim}}[\mathrm{T}]^2 + 2k_{\mathrm{r,dim}}[\mathrm{T_2}], \\
\frac{\mathrm{d}[\mathrm{T_2}]}{\mathrm{d}t} &= -k_{\mathrm{f,rep}}[\mathrm{D_G}][\mathrm{T_2}] + k_{\mathrm{r,rep}}[\mathrm{D_G{:}T_2}] - 2k_{\mathrm{f,dim}}[\mathrm{T}]^2 + 2k_{\mathrm{r,dim}}[\mathrm{T_2}], \\
\frac{\mathrm{d}[\mathrm{D_G{:}T_2}]}{\mathrm{d}t} &= k_{\mathrm{f,rep}}[\mathrm{D_G}][\mathrm{T_2}] - k_{\mathrm{r,rep}}[\mathrm{D_G{:}T_2}],
\end{aligned} \tag{7}$$

with initial conditions

$$x_{0,j} = \begin{pmatrix} [\mathrm{D_G}] \\ [\mathrm{D_T}] \\ [\mathrm{G}] \\ [\mathrm{T}] \\ [\mathrm{T_2}] \\ [\mathrm{D_G{:}T_2}] \end{pmatrix}(0) = \begin{pmatrix} 5 \text{ nM} \\ (\mathrm{D_{T0}})_j \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

and output

$$y_{i,\mathrm{test},j}(t) = [\mathrm{G}](t).$$

The process specific parameters are $\theta_{p,\mathrm{test}} = (k_{\mathrm{fG}}, k_{\mathrm{rG}}, k_{\mathrm{fT}}, k_{\mathrm{rT}}, k_{\mathrm{f,rep}}, k_{\mathrm{r,rep}}, k_{\mathrm{f,dim}}, k_{\mathrm{r,dim}})$, and for extract i, the environment specific parameters are $\theta_{e,i,\mathrm{cal}} = (\mathrm{Enz}_{0,i}, k_{\mathrm{c},i})$. As was the case in Example 1, we eliminated [Enz] from the state variables, and in doing so, removed the enzyme initial concentration $\mathrm{Enz}_{i,0}$, and hence the dependence on parameters, from the initial condition vector, $x_{0,j}$.

With these definitions, solving the data correction problem involves finding a method for predicting the behavior of the test circuit in the reference extract given measurements of its behaviour in the candidate extract, along with the calibration data, and calibration and test circuit models (see Figure 2). ◇

4.2 The Calibration-Correction Method as a Solution to the Data Correction Problem

In this section, we define the calibration-correction method, and apply it to the example of genetic circuit variability reduction described in Example 2. We begin by defining parameter identification as a set-valued operation on a data-model pair, which allows for multiple parameter points to fit the model to the data.

Definition 5 (Parameter Identification)**.** Let the set Γ_θ be the set of all pairs $(y, M(\theta))$ for which there exists a parameter $\hat{\theta} \in \Omega$ such that $y = M(\hat{\theta})$. Let $\mathcal{P}(\Omega)$ be the power set of Ω. We define the *parameter identification* of the θ coordinates of the model M as an operation $\mathrm{ID}_\theta : \Gamma_\theta \to \mathcal{P}(\Omega)$, with $\mathrm{ID}_\theta(y, M(\theta)) = \{\hat{\theta} \in \Omega \mid y = M(\hat{\theta})\}$ ◇

In the definition above, we have explicitly included θ as a subscript to ID and Γ. This is useful because we also allow for two methods of identifying parameters only over some subset of parameter coordinates. The first such method (over θ_p, suppose) is to identify values over all the parameter coordinates, and then to project the resulting set down to the coordinates of interest. This will be denoted by $\mathrm{proj}_{\theta_p} \mathrm{ID}_\theta(y, M)$, where $\theta = (\theta_e, \theta_p)$, and proj_{θ_p} is the projection operator that projects the sets of parameters to the θ_p coordinates. The second method involves a conditional version of the parameter estimation operation, and is defined as follows.

Definition 6 (Conditional Identification)**.** Consider the partition $\theta = (\theta_a, \theta_b) \in \Omega \subset \mathbb{R}^{q_a} \times \mathbb{R}^{q_b}$. Let $\Gamma_{\theta_a|\theta_b=\tilde{\theta}_b} \triangleq \{(y, M) \mid \exists \theta_a : y = M(\theta_a, \tilde{\theta}_b)\}$. Then, we define the *conditional identification* operator as

$$\mathrm{ID}_{\theta_a|\theta_b=\tilde{\theta}_b} : \Gamma_{\theta_a|\theta_b=\tilde{\theta}_b} \to \mathcal{P}(\mathrm{proj}_{\theta_a} \Omega),$$

with

$$\mathrm{ID}_{\theta_a|\theta_b=\tilde{\theta}_b}(y, M(\theta_a, \theta_b)) = \{\hat{\theta}_a \in \mathrm{proj}_{\theta_a} \Omega \mid y = M(\hat{\theta}_a, \tilde{\theta}_b)\}.$$

◇

We unclutter the notation by abbreviating $\mathrm{ID}_{\theta_a|\theta_b=\tilde{\theta}_b}(y, M(\theta_a, \theta_a))$, to $\mathrm{ID}_{\theta_a}(y, M(\theta_a, \tilde{\theta}_b))$, and $\Gamma_{\theta_a|\theta_b=\tilde{\theta}_b}$ to Γ_{θ_a}.

Next, we define the calibration-correction method as a sequence of steps involving parameter identification and prediction. Along with stating each step of the method in terms of single parameter points trajectories, we also give descriptions of the sets of all such points and trajectories. The definitions of these sets allow for the investigation of whether the non-identifiable parameter sets can be treated as equivalence classes with respect to this method. In particular, in Section 5, we will derive a set of conditions for the method to work when *arbitrary* points in the parameter sets are picked at the various stages of the method. Figure 3 shows a schematic description of the three steps of this procedure.

Definition 7 (The Calibration-Correction Method)**.** Consider the data correction problem in the model universe. We define the *calibration-correction method* as a sequence of steps that takes as input the tuple $(M_{\mathrm{cal}}, M_{\mathrm{test}}, \overline{y}_{1,\mathrm{cal}}, \overline{y}_{2,\mathrm{cal}}, \overline{y}_{2,\mathrm{test}})$ and returns a prediction of the behavior of the test process in the reference environment, $\hat{y}_{1,\mathrm{test}}$. The steps are:

1. **Calibration Step.** Find environment specific parameters that fit the calibration model to corresponding data for each of the environments, while sharing a common estimate of the process specific parameter vector. That is, find $\hat{\theta}_{e1,\mathrm{cal}}$ and $\hat{\theta}_{e2,\mathrm{cal}}$ such that the tuple $(\hat{\theta}_{e1,\mathrm{cal}}, \hat{\theta}_{e2,\mathrm{cal}}, \hat{\theta}_{p,\mathrm{cal}})$ satisfies $\overline{y}_{1,\mathrm{cal}} = M_{\mathrm{cal}}(\hat{\theta}_{e1,\mathrm{cal}}, \hat{\theta}_{p,\mathrm{cal}})$ and $\overline{y}_{2,\mathrm{cal}} = M_{\mathrm{cal}}(\hat{\theta}_{e2,\mathrm{cal}}, \hat{\theta}_{p,\mathrm{cal}})$ for some $\hat{\theta}_{p,\mathrm{cal}}$. The set of all such environment specific parameter points is constructed as follows: define the set of all valid $(\hat{\theta}_{e1,\mathrm{cal}}, \hat{\theta}_{e2,\mathrm{cal}}, \hat{\theta}_{p,\mathrm{cal}})$ tuples

$$\tilde{\Theta}_{\mathrm{cal}} \triangleq \Big\{ (\theta_{e1}, \theta_{e2}, \theta_p) \, \Big| \, \overline{y}_{i,\mathrm{cal}} = M_{\mathrm{cal}}(\theta_{e,i}, \theta_p),\ i = 1, 2 \Big\},$$

and then define the environment specific parameter sets as

$$E_{i,\mathrm{cal}} \triangleq \mathrm{proj}_{\theta_{e,i}} \tilde{\Theta}_{\mathrm{cal}}, \qquad i = 1, 2. \tag{8}$$

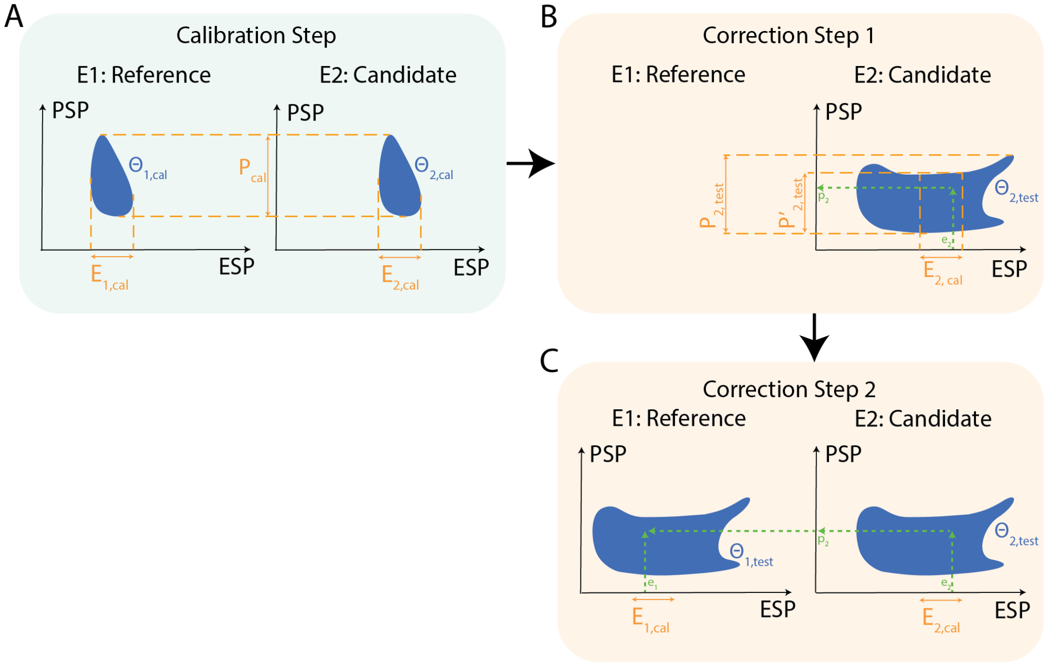

Figure 3: The calibration-correction method (Definition 7). ESP: environment specific parameters. PSP: process specific parameters. (A) Calibration step: fit the calibration model to calibration data in both environments jointly (sharing the process specific parameter coordinates), and project the resulting parameter set to the two sets of environment specific parameter coordinates to obtain the sets of calibration parameters for each environment, $E_{i,\text{cal}}$ for $i = 1, 2$. (B) Correction step 1: Estimate the test circuit process specific parameters ($\hat{\theta}_{p2,\text{test}}$), using the candidate environment ($\mathcal{E}_2$) test experiment data, the test process model, and an arbitrary point from the set of environment specific parameters associated with $\mathcal{E}_2$, denoted $\hat{\theta}_{e2,\text{cal}} \in E_{2,\text{cal}}$. (C) Correction step 2: Use an arbitrary point $\hat{\theta}_{e1,\text{cal}} \in E_{1,\text{cal}}$ from the set of reference extract environment specific parameters, along with the process specific parameter point estimated in correction step 1, $\hat{\theta}_{p2,\text{test}}$, in the test process model to generate a prediction of the reference extract trajectory, $\hat{y}_{1,\text{test}} = M_{\text{test}}\left(\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p2,\text{test}}\right)$.

2. **Correction Step One.** Identify process specific parameters of the test process in the candidate environment while holding the environment specific parameters at the value estimated at the previous step. That is, find $\hat{\theta}_{p2,\text{test}}$ such that $\overline{y}_{2,\text{test}} = M_{\text{test}}(\hat{\theta}_{e2,\text{cal}}, \hat{\theta}_{p2,\text{test}})$. The set of all such points is given by

$$P'_{2,\text{test}} \triangleq \bigcup_{\hat{\theta}_e \in E_{2,\text{cal}}} \text{ID}_{\theta_p|\theta_e=\hat{\theta}_e}\left(\overline{y}_{2,\text{test}}, M_{\text{test}}(\theta_e, \theta_p)\right). \tag{9}$$

3. **Correction Step Two.** Predict test process behavior in the reference environment using the process specific parameters estimated in the first correction step, and environment specific parameters estimated in the calibration step. That is, generate the prediction $\hat{y}_{1,\text{test}} = M_{\text{test}}(\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p2,\text{test}})$. The set of all predictions that can be generated is given by

$$Y_1 \triangleq \bigcup_{\hat{\theta}_e \in E_{1,\text{cal}}} \bigcup_{\hat{\theta}_p \in P'_{2,\text{test}}} \hat{y}_1(\hat{\theta}_e, \hat{\theta}_p), \tag{10}$$

where individual predictions are given by $\hat{y}_1(\hat{\theta}_e, \hat{\theta}_p) = M_{\text{test}}(\hat{\theta}_e, \hat{\theta}_p)$ for $\hat{\theta}_e \in E_{1,\text{cal}}$ and $\hat{\theta}_p \in P'_{2,\text{test}}$.

◇

Before describing this with an example, we describe how multiple experimental conditions (environments or initial conditions) may be handled.

Example 3 (Collecting Models and Parameter Consensus)**.** Consider once again the setup in Example 1, with two extracts $\mathcal{E}_1$ and $\mathcal{E}_2$ and five GFP DNA concentrations (Figure 2A, C). There are ten output trajectories—one for each combination of extract and DNA concentration—denoted $\overline{y}_{i,\text{cal},j}$ for the i-th extract and j-th initial condition, and ten corresponding models, $y_{i,\text{cal},j} = M_{\text{cal}}\left((\theta_{e,i,\text{cal}}, \theta_{p,\text{cal}}), x_{0,j}\right)$. The resulting data-model pair is a collection of these individual data trajectories and models, and involve a common set of shared parameters, as follows,

$$\left(\overline{y}_{\text{cal}}, M_{\text{cal}}\left(\theta_{e1,\text{cal}}, \theta_{e2,\text{cal}}, \theta_{p,\text{cal}}, \left(x_{0,1,1}, x_{0,1,2}, \ldots, x_{0,2,5}\right)\right)\right) \tag{11}$$

$$= \left\{ \begin{array}{c} \left(\overline{y}_{1,\text{cal},1}\right), M_{\text{cal}}\left((\theta_{e1,\text{cal}}, \theta_{p,\text{cal}}), x_{0,1,1}\right)\right), \\ \left(\overline{y}_{1,\text{cal},2}\right), M_{\text{cal}}\left((\theta_{e1,\text{cal}}, \theta_{p,\text{cal}}), x_{0,1,2}\right)\right), \\ \vdots \\ \left(\overline{y}_{1,\text{cal},5}\right), M_{\text{cal}}\left((\theta_{e1,\text{cal}}, \theta_{p,\text{cal}}), x_{0,1,5}\right)\right), \\ \left(\overline{y}_{2,\text{cal},1}\right), M_{\text{cal}}\left((\theta_{e2,\text{cal}}, \theta_{p,\text{cal}}), x_{0,2,1}\right)\right), \\ \vdots \\ \left(\overline{y}_{2,\text{cal},5}\right), M_{\text{cal}}\left((\theta_{e2,\text{cal}}, \theta_{p,\text{cal}}), x_{0,2,5}\right)\right) \end{array} \right\}. \tag{12}$$

Estimating $\theta_{e,i,\text{cal}} = (\text{Enz}_{0,i}, k_{\text{c},i})$, and $\theta_{p,\text{cal}} = (k_{\text{fG}}, k_{\text{rG}})$ from data, or predicting outputs given these parameters involves treating their repeated occurrences in equation (12) as the same parameters used multiple times. As mentioned earlier, this process of sharing parameters across related models is called hard parameter sharing or parameter consensus.

Note that the more general case of multiple different calibration circuits is handled via a straightforward extension of the idea of collecting corresponding models and allowing for parameter sharing across them. ◇

Next, we illustrate the calibration-correction method on the data in the our case study. The overall methodology is illustrated in the schematic in Figure **??**.

Example 4 (The Calibration-Correction Method for Genetic Circuit Variability Reduction)**.** In this example, we apply the calibration-correction method to solve the data correction problem in Example 2, where our objective is to predict the behavior of a given test circuit in a reference extract, given its behavior in a candidate extract.

The data and models we use in the calibration step are those described in Example 3. In the calibration step (Figure **??**A-B), we estimate all points $\theta = \left(\theta_{e1,\text{cal}}, \theta_{e2,\text{cal}}, \theta_{p,\text{cal}}\right) = \left(\text{Enz}_{0,1}, k_{\text{c},1}, \text{Enz}_{0,2}, k_{\text{c},2}, k_{\text{fG}}, k_{\text{rG}}\right) \in \tilde{\Theta}_{\text{cal}} \subset \mathbb{R}_+^6$ that simultaneously fit the ten models (two extracts and five initial DNA concentrations) to their corresponding data trajectories, with the parameter sharing scheme described in Example 3.

An estimate of the set $\tilde{\Theta}_{\text{cal}}$ may be constructed using Bayesian inference via Markov chain Monte Carlo (MCMC), which samples from the posterior distribution of parameters $\mathbb{P}(\theta_{e1,\text{cal}}, \theta_{e2,\text{cal}}, \theta_{p,\text{cal}} \mid \overline{y}_{\text{cal}}, M_{\text{cal}})$ by drawing parameter points from a prior distribution,[2] and accepting points probabilistically, depending on how far the corresponding model predictions are from observed data.[3, 4] Figure 4B shows pairwise projections of the set of parameters ($\tilde{\Theta}_{\text{cal}}$) computed using MCMC, with parameter sharing as described in Example 3 above, and fits shown in Figure 4A.

The model, parameters and initial conditions for the test circuit were described in detail in Example 2. Recall that the first correction step involves estimating its process specific parame-

[2] We use a uniform distribution within a large hypercube in parameter space as our prior—the so-called uninformative prior.

[3] See Singhal et al. [14] for a toolbox that enables MCMC with hard parameter sharing.

[4] Approximate Bayesian computation is another tool for computing posterior distributions of parameters, and may be used to approximate $\tilde{\Theta}_{\text{cal}}$. See also Hori and Murray [15] for yet another tool.

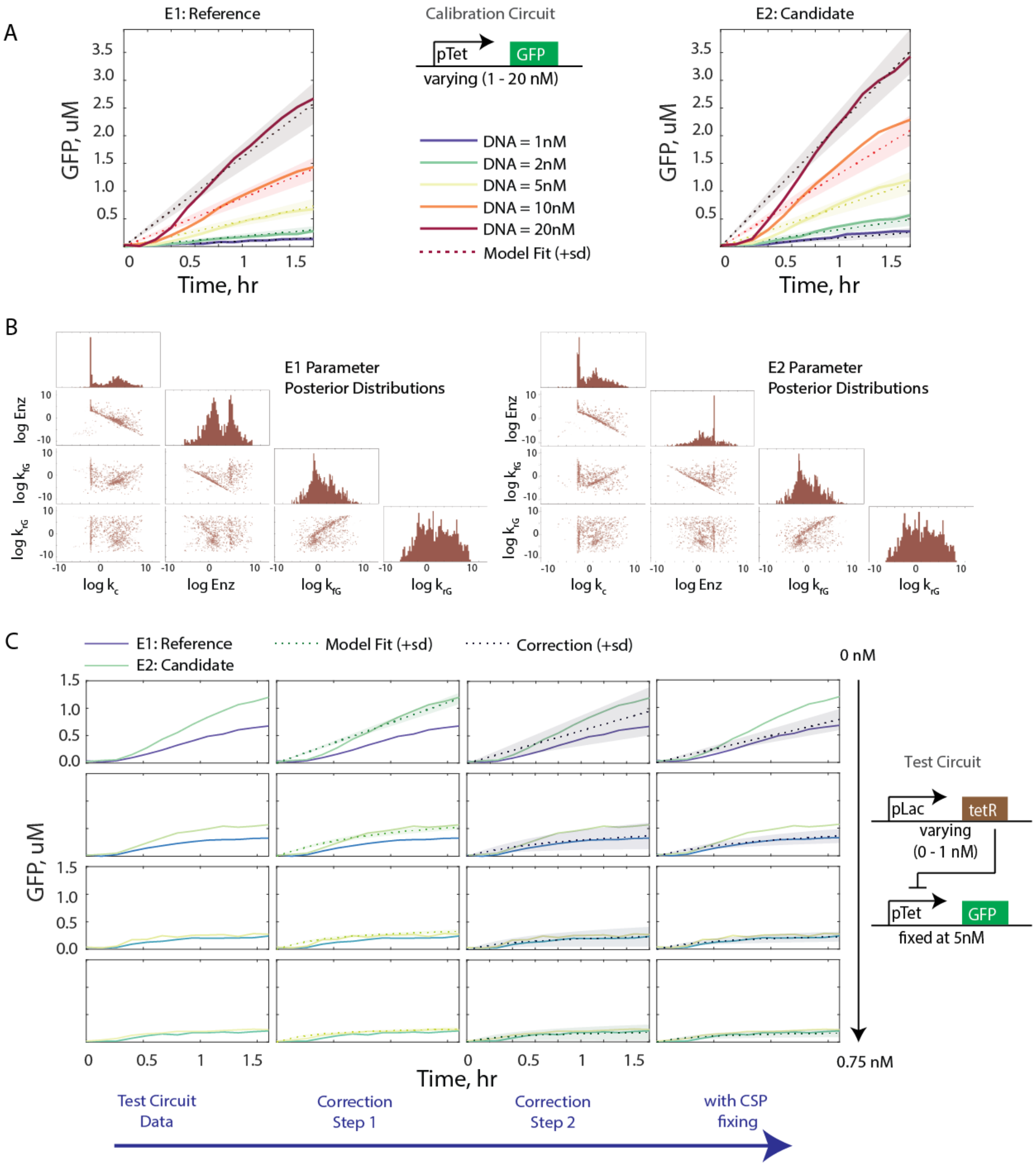

Figure 4: The calibration-correction method (Definition 7) on the data shown in Figure 2. (A) Calibration step model fits. Solid lines: experimental data for the constitutive expression circuit measured in two extracts, $\mathcal{E}_1$ and $\mathcal{E}_2$. Dotted lines and shaded region: mean and standard deviations of 500 trajectories simulated using point drawn from the posterior distribution, which approximates Θ_{cal}. (B) Pairwise projections of the posterior distribution, split into two corner plots corresponding to the two extracts. (C) Correction steps 1 and 2 on the TetR repression test circuit data. Solid lines: experimental data. Purple: reference extract $\mathcal{E}_1$ trajectories. Green: candidate extract $\mathcal{E}_2$ trajectories. Rows of subplots correspond to different initial tetR DNA concentrations (initial conditions). Dotted lines and shaded regions: mean and standard deviations of simulated trajectories (using parameters drawn from the relevant distributions, see main text). The columns, starting from the left, are: test circuit data in the two extracts, correction step one, the second correction step, and the improvement from applying process specific parameter conditioning to the calibration-correction method.

ters $\theta_{p,\text{test}} = (k_{\text{fG}}, k_{\text{rG}}, k_{\text{fT}}, k_{\text{rT}}, k_{\text{f,rep}}, k_{\text{r,rep}}, k_{\text{f,dim}}, k_{\text{r,dim}}) \in P_{2,\text{test}}$ by fitting the test model behavior $\hat{y}_{2,\text{test}} = M_{\text{test}}\big((\theta_{e2}, \theta_{p,\text{test}}), x_0\big)$ to the test circuit's behavior in the candidate extract (over all initial conditions simultaneously, see Example 3), while holding its environment specific parameters at an arbitrary point in the candidate extract's environment specific parameter set, $\theta_{e2,\text{cal}} = (\text{Enz}_{0,2}, k_{\text{c},2}) \in E_{2,\text{cal}}$.

The model fits are shown as dotted green curves and shaded regions (mean, s.d.) in the second column in Figure 4C. Fixing the environment specific parameters to a point in $E_{1,\text{cal}}$ and drawing 500 points from $P_{2,\text{test}}$ to generate the corrected trajectories implements correction step two, and the results are shown as the purple dotted curve and shaded region in the third column in Figure 4C. We see in the third column that the purple dotted curves move closer to the ground truth (solid purple lines), but the variance in the predicted trajectories increases, possibly due to parameter covariation (described in Section 6). Process specific parameter conditioning (see Definition 13 and Proposition 2 below) at the calibration step helps to correct for this, and the resulting predicted reference extract trajectories are shown as purple dotted lines (and shaded regions) in the fourth column

We compute the degree of variability reduction achieved by our procedure on this test circuit data. We define two metrics to measure the variability reduction. Both metrics take values in $[0, \infty)$, with a value of 0 corresponding to perfect correction, a value of 1 corresponding to no correction on average, and values larger than 1 corresponding to the predicted trajectories being further from the true reference trajectories than the original candidate trajectories are.

The first metric measures the the ratio of the sum of the deviations between the corrected and reference trajectories to the sum of the deviations between the original reference and candidate trajectories. We can write the metric as,

$$R_1 = \frac{\sum_{j=1}^{n_{IC}} ||\hat{y}_{1,\text{test},j} - \overline{y}_{1,\text{test},j}||_2}{\sum_{j=1}^{n_{IC}} ||\overline{y}_{2,\text{test},j} - \overline{y}_{1,\text{test},j}||_2},$$

where the sum is taken over the n_{IC} experimental conditions (which, in this case, are the first four tetR DNA concentrations, as shown in Figure 4). For our dataset, we compute this value to be $R_1 = 0.42$.

The second metric computes, summed over the n_{IC} initial conditions, the ratio of the deviation between the corrected trajectory and the reference extract trajectory, and the deviation between the original candidate extract trajectory and the reference extract trajectory. It then takes the mean of these individual ratios to give a score for the average correction. It is defined as

$$R_2 = \frac{1}{n_{IC}} \sum_{j=1}^{n_{IC}} \frac{||\hat{y}_{1,\text{test},j} - \overline{y}_{1,\text{test},j}||_2}{||\overline{y}_{2,\text{test},j} - \overline{y}_{1,\text{test},j}||_2},$$

and gives a value of 0.48 when computed for our dataset.

◇

The version of the calibration step defined above is straightforward to implement computationally (using inference tools that allow for hard parameter sharing between models, see for instance the consensus MCMC tools described in Singhal et al. [14]), since the sets $E_{i,\text{cal}}$, for $i = 1, 2$, are simple projections computed from the estimated set.

We also give an equivalent definition here that allows for the estimation of the parameters for the two extracts separately, followed by a restriction procedure that enforces agreement between the process specific parameters estimated in the two extracts. We start with estimating the environment- and process specific parameter sets for individual extracts, $\Theta_{i,\text{cal}} \triangleq \text{ID}_\theta\left(\overline{y}_{i,\text{cal}}, M_{\text{cal}}(\theta)\right)$, $i = 1, 2$, and then compute the set of process specific parameters where these agree, $P_{\text{cal}} \triangleq \text{proj}_{\theta_p} \Theta_{1,\text{cal}} \cap \text{proj}_{\theta_p} \Theta_{2,\text{cal}}$. Finally, the environment specific parameter sets are generated by restricting both $\Theta_{i,\text{cal}}$ by P_{cal},

$$E_{i,\text{cal}} \triangleq \left\{ \theta_e \;\middle|\; \exists \theta_p \in P_{\text{cal}} : (\theta_e, \theta_p) \in \Theta_{i,\text{cal}} \right\}, \qquad i = 1, 2.$$

The fact that the sets $\Theta_{i,\text{cal}}$, $i = 1, 2$, are estimated separately can be useful in cases where the dimension of the spaces θ_e and θ_p live in (that is, q_E and q_P) are large enough that estimating

$\tilde{\Theta}_{\text{cal}} \in \mathbb{R}^{2q_E+q_P}$ might be much more difficult compared to $\Theta_{i,\text{cal}} \in \mathbb{R}^{q_E+q_P}$. The trade-off here is that intersections and restrictions of sets represented by point clouds can be computationally difficult in high dimensional spaces. Finally, the lemma in Appendix B.1 establishes the equivalence of this definition to the one given in equation (8).

Note that the set $P'_{2,\text{test}}$ in Definition 7 is a subset of the larger set

$$P_{2,\text{test}} \triangleq \text{proj}_{\theta_p} \text{ID}_\theta(\overline{y}_{2,\text{test}}, M_{\text{test}}).$$

Indeed, $P'_{2,\text{test}}$ is obtained from $P_{2,\text{test}}$ by only keeping the points whose corresponding θ_e coordinate values were in the calibration set $E_{2,\text{cal}}$. We use $P'_{2,\text{test}}$ because in the first correction step, we identify θ_p only after fixing the value of θ_e to an arbitrary point within $E_{2,\text{cal}}$.

5 Identifiability Conditions

In this section, we show that structural global identifiability is not necessary for the calibration-correction method to solve the data correction problem. This will be stated as a corollary of the main result of this section (Theorem 1), which gives conditions on the non-identifiable parameters under which the calibration-correction method solves the data correction problem. The main idea is that since correction only needs to be applied to the output variables, and not to the full state vector, the parameters only need to be ‘identifiable enough’ to reconstruct the output trajectories.

This notion is closely related to the sets of output-indistinguishable parameters being equivalence classes with respect to the initial conditions, inputs and outputs of a model. While these sets may be equivalence classes with respect to individual data-model pairs, some additional restrictions need to be placed on these sets if they are to be treated as equivalence classes with respect to the calibration-correction method.

5.1 The Model Universe and Failure Conditions

Our analytical results will be stated and proved in a virtual *model universe*, where artificial data $\overline{y}$ are generated using nominal models $\overline{M}$ with known nominal parameter values $\overline{\theta}$. That is, in the model universe, we identify $\mathcal{H} = (\mathcal{S}, x_0, \overline{y})$ with $\overline{y} = \overline{M}(\overline{\theta}, x_0)$.

We will also make a *model correctness* assumption, denoted $M = \overline{M}$, which states that in the model universe, the models we use to estimate parameters from the data are the very models used to generate the data. In particular, both models will have the same dynamical equations, f, and output functions, h, specifying them. Working in a model universe, along with this additional correctness assumption, allows us to look at the interaction of non-identifiability with our method in isolation, that is, without also having to be concerned with whether our models are good models of the system that generated the data. Important issues associated with model correctness or the use of approximate models (that arise due to model-order reduction, for instance) are left as future work. Furthermore, it is worth explicitly stating that even though a single nominal parameter point is used to generate the output trajectory, the non-identifiability of parameters when their identification is attempted using the output trajectory and the nominal models arises because of the structure of the dynamics function f and that of the output function h. Thus, when stating and proving our main results in Section 5, we will always use single points to specify nominal parameter values, even when we can only identify sets of parameter values from the output trajectories.

The identifiability results in this section are developed in the context of this idealized model universe. In this setting, there are two places at which the calibration-correction method can fail. Avoiding these *failure conditions* forms the basis of our proofs below.

Definition 8 (Failure Condition 1). *Failure condition one* occurs if a parameter identification step is attempted when no parameter exists such that the model fits the data. This means that the data-model pair (y, M) under consideration is not in the domain Γ of the operator ID. In terms of the set based formulation of the calibration-correction method, this failure condition occurs if it occurs for *any* point in the set. ⋄

For example, if it is possible to find an $\hat{\theta}_{e2,\text{cal}}$ in the calibration step such that in the first correction step, there is no $\tilde{\theta_p}$ that satisfies $\overline{y}_{2,\text{test}} = M_{\text{test}}(\hat{\theta}_{e2,\text{cal}}, \tilde{\theta_p})$, then the parameter estimation step fails at this point. In terms of the set based formulation of equation (9), this failure condition occurs if it occurs for *any* point θ_e in $E_{2,\text{cal}}$.

Definition 9 (Failure Condition 2). *Failure condition two* occurs if correction step two is able to produce a trajectory not equal to the true trajectory, that is, $\hat{y}_{1,\text{test}} \neq \overline{y}_{1,\text{test}}$. In terms of the set Y_1 defined in equation (10), this means that Y_1 contains at least one element that is not equal to $\overline{y}_{1,\text{test}}$. ◇

This condition occurs when the calibration-correction method constructs parameter points not lying on the true parameter manifold. This can happen in, for instance, correction step two, where we combine parameter coordinates from $E_{1,\text{cal}}$ and $P'_{2,\text{test}}$ to generate a prediction. As we see, this is useful for showing contradiction in the results in this section and the next.

As noted in Remark 2, with real data, this idealized setting must be relaxed to account for noise in the data. This can be done by replacing equality with some notion of nearness. While we consider noisy data in the examples in this study, their theoretical investigation involves notions of practical identifiability, and is beyond the scope of the current study.

Theorem 1 (Parameter consistency). *Consider the data correction problem under the model universe assumption, the calibration-correction method, and the sets $\tilde{\Theta}_{\text{cal}}$, $E_{1,\text{cal}}$, $E_{2,\text{cal}}$ and $P'_{2,\text{test}}$ as defined in Definition 7. Define $\Theta_{i,\text{test}} \triangleq \text{ID}_{\theta}\left(\overline{y}_{i,\text{test}}, \overline{M}_{\text{test}}(\theta)\right)$ for $i = 1, 2$. Then, the conditions,*

$$\tilde{\Theta}_{\text{cal}} \neq \emptyset, \tag{13}$$

$$E_{2,\text{cal}} \subseteq \text{proj}_{\theta_e} \Theta_{2,\text{test}}, \tag{14}$$

$$E_{1,\text{cal}} \times P'_{2,\text{test}} \subseteq \Theta_{1,\text{test}}, \tag{15}$$

are necessary and sufficient for the calibration-correction method to solve the data correction problem.

Proof. See Appendix A □

We may give some physical interpretations of the conditions (13-15). To do this, we first note that condition (15) implies (see Lemma 3 in Appendix B.2)

$$E_{1,\text{cal}} \subseteq \text{proj}_{\theta_e} \Theta_{1,\text{test}}, \tag{16}$$

$$P'_{2,\text{test}} \subseteq P'_{1,\text{test}}, \tag{17}$$

where $P'_{1,\text{test}}$ is defined in a similar way to $P'_{2,\text{test}}$,

$$P'_{1,\text{test}} \triangleq \bigcup_{\hat{\theta}_e \in E_{1,\text{cal}}} \text{ID}_{\theta_p|\theta_e=\hat{\theta}_e}\left(\overline{y}_{1,\text{test}}, M_{\text{test}}\big(\theta_e, \theta_p\big)\right).$$

Conditions (13) and (16) may be interpreted to mean that the calibration experiments must be more informative about the environment specific parameters than the test circuit experiments. This follows from the fact that the sets of output-indistinguishable environment specific parameters obtained from the calibration step are subsets of the corresponding sets from the test circuits, $\text{proj}_{\theta_e} \Theta_{i,\text{test}}$.

Condition (17) says that the process specific parameter sets for the test circuit estimated in each extract, if estimated by first fixing the environment specific parameters to values obtained at the calibration stage, must agree. Agreement here is defined to be unidirectional, with one set being a subset of another. This is only because the correction being performed is from the candidate extract to the reference extract. If bidirectional correction (Corollary 2, below) were required, then we would have equality in Condition (17).

Finally, Condition (15) says that the environment- and process specific parameter coordinates in the set $\Theta_{1,\text{test}}$ can only *covary* outside $E_{1,\text{cal}} \times P'_{2,\text{test}}$, that is, all the points within this set must belong to $\Theta_{1,\text{test}}$. Covariation is defined in Section 6.

Corollary 1 (Sufficiency of Structural Global Identifiability)**.** *Structurally globally identifiable models are sufficient for the calibration-correction method to solve the data correction problem in the model universe.*

Proof. See Appendix A □

Corollary 2 (Bidirectional Correction)**.** *To be able to correct the test data from either extract to the other requires that:*

$$\begin{aligned}\tilde{\Theta}_{\text{cal}} &\neq \emptyset,\\ E_{i,\text{cal}} &\subseteq \operatorname{proj}_{\theta_e} \Theta_{i,\text{test}}, \qquad i = 1, 2,\\ E_{1,\text{cal}} \times P'_{2,\text{test}} &\subseteq \Theta_{1,\text{test}},\\ E_{2,\text{cal}} \times P'_{1,\text{test}} &\subseteq \Theta_{2,\text{test}}.\end{aligned}$$

Proof. The proof is a simple union of the sets of conditions implied by Theorem 1 for each direction of correction. □

In the bidirectional scenario, the parameter 'agreement' condition $P'_{2,\text{test}} \subseteq P'_{1,\text{test}}$ discussed in the text following Theorem 1 gets transformed into $P'_{2,\text{test}} = P'_{1,\text{test}}$.

Next we discuss the case of correcting the calibration data itself. This will be important in the next section when we examine the effect of a phenomenon called parameter covariation on the calibration-correction method. There, we will prove that a modified version of the method is able to solve the problem at least for this case, even in the presence of parameter covariation.

Corollary 3 ('Test = Calibration' Case)**.** *Consider the data correction problem for the case where the test data and models are the same as the calibration data and models, that is,* $\overline{y}_{i,\text{test}} = \overline{y}_{i,\text{cal}}$ *and* $\overline{M}_{\text{test}} = \overline{M}_{\text{cal}}$ *for* $i = 1, 2$*. Furthermore, let* $\Theta_{i,\text{cal}} \triangleq \text{ID}_\theta\left(\overline{y}_{i,\text{cal}}, M_{\text{cal}}(\theta)\right)$ *for* $i = 1, 2$*, and*

$$P'_{2,\text{cal}} \triangleq \bigcup_{\tilde{\theta}_e \in E_{2,\text{cal}}} \text{ID}_{\theta_p}\left(\overline{y}_{2,\text{cal}}, M_{\text{cal}}\left(\tilde{\theta}_e, \theta_p\right)\right).$$

Then, the conditions

$$\tilde{\Theta}_{\text{cal}} \neq \emptyset, \tag{18}$$

$$E_{2,\text{cal}} \subseteq \operatorname{proj}_{\theta_e} \Theta_{2,\text{cal}}, \tag{19}$$

$$E_{1,\text{cal}} \times P'_{2,\text{cal}} \subseteq \Theta_{1,\text{cal}}, \tag{20}$$

are necessary and sufficient for the calibration-correction method to solve this problem.

Proof. Simply specialize the conditions in Theorem 1 to this case. □

6 Environment- and Process Specific Parameter Covariation Causes the Calibration-Correction Method to Fail

In this section, we describe covariation, and show that it causes the calibration-correction method to fail. We then discuss an improvement to the method that addresses this issue. We start with defining a device that will be useful for taking slices of parameter sets.

Definition 10 (Cutting Plane)**.** Consider the space of parameters $\mathbb{R}^q$, the vector $\theta \in \mathbb{R}^q$ partitioned into two sets of coordinates $\theta = (\theta_a, \theta_b) \in \mathbb{R}^{q_a} \times \mathbb{R}^{q_b}$ and the subspaces $A \triangleq \mathbb{R}^{q_a} \times \{0\}$ and $B \triangleq \{0\} \times \mathbb{R}^{q_b}$ corresponding to the θ_a and θ_b coordinates respectively. Let $\tilde{\theta}_a \in A$. Then, we denote the *cutting plane* generated by shifting the origin of B to $(\tilde{\theta}_a, 0)$ with the notation $\text{cut}_{\theta_b}(\tilde{\theta}_a)$. ◇

Definition 11 (Parameter Covariation)**.** Consider the space of parameters $\mathbb{R}^q$ and the vector $\theta \in \mathbb{R}^q$ partitioned into two sets of coordinates $\theta = (\theta_a, \theta_b) \in \mathbb{R}^{q_a} \times \mathbb{R}^{q_b}$. Consider some set of parameters $\Theta \subseteq \mathbb{R}^q$. If there exist $\tilde{\theta}_{a1}, \tilde{\theta}_{a2} \in \operatorname{proj}_{\theta_a} \Theta$ such that $\operatorname{proj}_{\theta_b}\left(\Theta \cap \text{cut}_{\theta_b}(\tilde{\theta}_{a1})\right) \neq \operatorname{proj}_{\theta_b}\left(\Theta \cap \text{cut}_{\theta_b}(\tilde{\theta}_{a2})\right)$, then Θ is said to have *parameter covariation* of its θ_b coordinates with respect to its θ_a coordinates. ◇

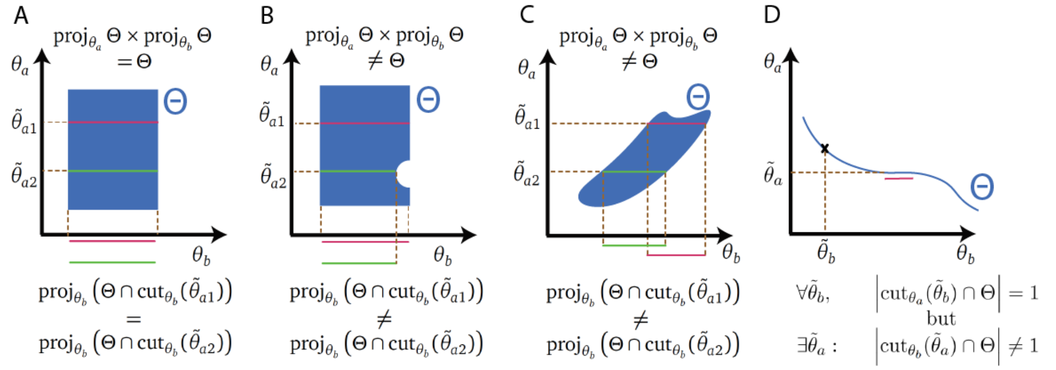
A
$\mathrm{proj}_{\theta_a}\Theta \times \mathrm{proj}_{\theta_b}\Theta$
$=\Theta$
θ_a
$\tilde{\theta}_{a1}$
$\tilde{\theta}_{a2}$
Θ
θ_b
$\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a1})\right)$
$=$
$\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a2})\right)$
B
$\mathrm{proj}_{\theta_a}\Theta \times \mathrm{proj}_{\theta_b}\Theta$
$\neq\Theta$
θ_a
$\tilde{\theta}_{a1}$
$\tilde{\theta}_{a2}$
Θ
θ_b
$\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a1})\right)$
$\neq$
$\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a2})\right)$
C
$\mathrm{proj}_{\theta_a}\Theta \times \mathrm{proj}_{\theta_b}\Theta$
$\neq\Theta$
θ_a
$\tilde{\theta}_{a1}$
$\tilde{\theta}_{a2}$
Θ
θ_b
$\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a1})\right)$
$\neq$
$\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a2})\right)$
D
θ_a
$\tilde{\theta}_a$
Θ
$\tilde{\theta}_b$
θ_b
$\forall\tilde{\theta}_b, \quad \left|\mathrm{cut}_{\theta_a}(\tilde{\theta}_b) \cap \Theta\right| = 1$
but
$\exists\tilde{\theta}_a: \quad \left|\mathrm{cut}_{\theta_b}(\tilde{\theta}_a) \cap \Theta\right| \neq 1$

Figure 5: Covariation and 'thin' covariation. (A) The absence of covariation in a set of parameters corresponds to a Cartesian product condition being met (see Lemma 1). (B) An example of a 'minimal' violation in the covariation condition of Definition 11. (C) In general, covariation will have more general deviations in shape from the 'rectangular' shape implied by the Cartesian product. See Section 8 and Figure 7 for examples of covariation found in mass-action kinetics models used in synthetic biology. (D) The covariation in the θ_a coordinates of Θ is 'thin' with respect to the θ_b coordinates (Definition 12). The definition of thin covariation is directional: here, the covariation in the θ_b coordinates is not thin with respect to the θ_a coordinates, due to the defining condition being violated at the region marked in red (that is, at $\tilde{\theta}_a$).

We will often abbreviate parameter covariation to just covariation, and say that parameter coordinates can *covary*.

Lemma 1. *Let $\theta = (\theta_a, \theta_b) \in \Theta \subseteq \mathbb{R}^q$ be a partition of the coordinates of $\mathbb{R}^q$. Then, the set Θ has covariation of its θ_b coordinates with respect to its θ_a coordinates if and only if* $\mathrm{proj}_{\theta_a}\Theta \times \mathrm{proj}_{\theta_b}\Theta \neq \Theta$.

Proof. See Appendix A □

Corollary 4. *The set Θ has covariation of its θ_b coordinates with respect to its θ_a coordinates if and only if it has covariation of its θ_a coordinates with respect to its θ_b coordinates.*

Proof. The proof of Lemma 1 can be repeated with straightforward modifications (essentially swapping the roles of θ_a and θ_b) to show the equivalence of the condition $\mathrm{proj}_{\theta_a}\Theta \times \mathrm{proj}_{\theta_b}\Theta \neq \Theta$ to the set Θ having covariation of its θ_a coordinates with respect to its θ_b coordinates. □

This equivalence will allow us to refer to sets having covariation with respect to a given partition. Specifically, we will consider Θ having covariation with respect to the (θ_e, θ_p) partition.

Next, we show that in the presence of covariation, the calibration-correction method is unable to solve the data correction problem even in the case when the test data are the calibration data themselves. In particular, we will assume that the restriction of $\Theta_{1,\mathrm{cal}}$ to $E_{1,\mathrm{cal}} \times \mathrm{proj}_{\theta_p}\Theta_{2,\mathrm{cal}}$ has covariation with respect to the (θ_e, θ_p) partition.

Proposition 1. *Consider the 'Test = Calibration' case of the data correction problem described in Corollary 3, along with the definitions of the various sets given there. Assume the conditions*

$$\tilde{\Theta}_{\mathrm{cal}} \neq \emptyset, \tag{21}$$

$$P'_{2,\mathrm{cal}} \subseteq \mathrm{proj}_{\theta_p}\Theta_{1,\mathrm{cal}}, \tag{22}$$

$$E_{i,\mathrm{cal}} \subseteq \mathrm{proj}_{\theta_e}\Theta_{i,\mathrm{cal}}, \qquad i = 1, 2, \tag{23}$$

hold, but the set

$$\Theta'_{1,\mathrm{cal}} \triangleq \Theta_{1,\mathrm{cal}} \cap \left(E_{1,\mathrm{cal}} \times \mathrm{proj}_{\theta_p}\Theta_{2,\mathrm{cal}}\right)$$

has covariation in its θ_e coordinates with respect to its θ_p coordinates. Then, the calibration-correction method fails to solve this problem.

Proof. See Appendix A □

7 A Modification via Process Specific Parameter Conditioning Addresses Covariation

Next, we define a specific type of covariation, which we call *thin* covariation, and show that a modification to the calibration-correction method is able to solve the data correction problem for the 'Test = Calibration' case when the process specific parameter coordinates covary in this way with respect to the environment specific parameter coordinates. In Section 8.1, we will show that even the simplest models show non-identifiability with this type of covariation.

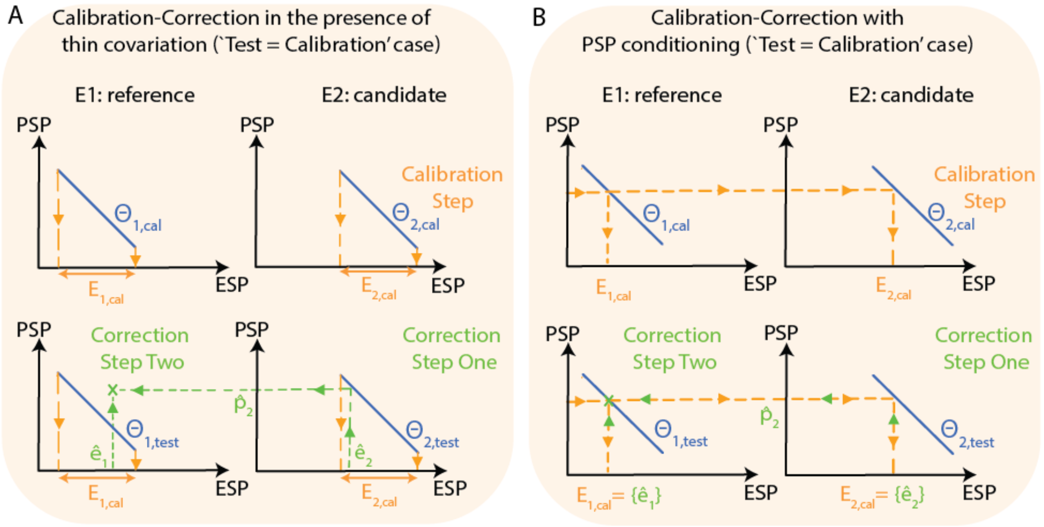

Figure 6: (A) Intuition behind the proofs of how thin covariation causes a failure of the calibration-correction method in the 'Test = Calibration' case, and how process specific parameter conditioning restores its ability to solve the data correction problem (Propositions 1 and 2).

Definition 12 (Thin Covariation)**.** Let $\Theta \subset \mathbb{R}^q$ be a set of parameters and let $(\theta_a, \theta_b) \in \mathbb{R}^q$ be a partition of the coordinates of $\mathbb{R}^q$. If Θ covaries with respect to this partition and if for all $\tilde{\theta}_b \in \mathrm{proj}_{\theta_b}\Theta$, we have $\left|\mathrm{cut}_{\theta_a}(\tilde{\theta}_b) \cap \Theta\right| = 1$, then we say that the covariation of the θ_a coordinates of Θ is *thin* with respect to the θ_b coordinates. ◇

We note that if $\Theta \triangleq \mathrm{ID}_\theta(\overline{y}, M(\theta))$, then the condition that for all $\tilde{\theta}_b \in \mathrm{proj}_{\theta_b}\Theta$, we have $\left|\mathrm{cut}_{\theta_a}(\tilde{\theta}_b) \cap \Theta\right| = 1$ is equivalent to the θ_a coordinates of the model $M(\theta_a, \theta_b)$ being structurally globally identifiable for each fixed θ_b. Thus we see that this type of covariation is essentially a statement about the some coordinates being conditionally structurally globally identifiable, despite covarying with respect to the remaining coordinates.

Definition 13 (Process Specific Parameter Conditioning)**.** Consider the sets $\Theta_{i,\mathrm{cal}} \triangleq \mathrm{ID}_\theta\left(\overline{y}_{i,\mathrm{cal}}, M_{\mathrm{cal}}(\theta)\right)$, $i = 1, 2$ and let $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p}\Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p}\Theta_{2,\mathrm{cal}}$. Then, we define *process specific parameter conditioning* as a modification to the calibration step in which the sets $E_{i,\mathrm{cal}} \triangleq \mathrm{proj}_{\theta_e}\left(\mathrm{cut}_{\theta_e}\left(\tilde{\theta}_p\right) \cap \Theta_{i,\mathrm{cal}}\right)$ for $i = 1, 2$. ◇

Proposition 2. *Consider the sets* $\Theta_{i,\mathrm{cal}} \triangleq \mathrm{ID}_\theta\left(\overline{y}_{i,\mathrm{cal}}, M_{\mathrm{cal}}(\theta)\right)$ *for* $i = 1, 2$, *and the partition* $\theta = \left(\theta_e, \theta_p\right)$. *Assume that the* $\Theta_{i,\mathrm{cal}}$ *have thin covariation in their* θ_p *coordinates with respect to their* θ_e *coordinates. Then, the calibration-correction method with process specific parameter conditioning is able to solve the data correction problem for the 'Test = Calibration' case of Corollary 3.*

Proof. See Appendix A □

8 Computational Investigation of Covariation and Process Specific Parameter Conditioning

In this section we investigate the effect of covariation on the calibration-correction method computationally, and show that process specific parameter conditioning helps reduce the error introduced by such covariation.

The approach will be to generate artificial data using the models in Equations (2) and (6) with a fixed set of parameters, and then to use these same models to perform the calibration-correction method. In this way, we implement the model universe and model correctness assumptions of Section 5.1, enabling us to study the effects of structural non-identifiability in isolation.

8.1 The 'Test = Calibration' case of Corollary 3

First, we demonstrate that even the simplest models show non-identifiability and thin covariation in their process specific parameter coordinates with respect to their environment specific parameter coordinates, and this causes the calibration-correction method to fail to correct even the calibration data ('Test = Calibration' case). We also show that with process specific parameter conditioning, this is avoided due to the mechanics of the proof of Proposition 2.

We begin by generating artificial calibration data for extracts $\mathcal{E}_1$ and $\mathcal{E}_2$ using the constitutive expression circuit (Figure 7A), with the model in equation (2), and the parameters in Table **??** in Appendix C. The process specific parameters used were the same for the models in both extracts, while the environment specific parameters differed for the two extracts. The true trajectories are shown as dotted curves in Figure 7B, C. We have added a small amount of noise to these for easier visualization of overlapping trajectories; however the trajectories used as model universe data in the calibration-correction method do not contain this added noise. The calibration step was performed with $k_{\mathrm{fG}} = 5$ fixed at its true value, reducing the number of parameters in the model to three (process specific parameter k_{rG}, and environment specific parameters $[\mathrm{Enz}]$ and k_{c}), allowing for the visualization of the joint distribution of the parameter samples that result from performing MCMC. This visualization is the most direct method of seeing the existence of non-identifiability and of thin covariation in the parameters.

The fitting of the model to the data (Figure 7B, C) in the calibration step results in an estimate of the joint distribution of the parameter vector $(\theta_{e1}, \theta_{e2}, \theta_p) \in \tilde{\Theta}_{\mathrm{cal}}$. The three dimensional scatter plots of empty blue circles in Figure 7D, E show the results of this estimation marginalized to the coordinates $(\theta_{e1}, \theta_p) = ([\mathrm{Enz}]_1, k_{\mathrm{c}1}, k_{\mathrm{rG}})$ and $(\theta_{e2}, \theta_p) = ([\mathrm{Enz}]_2, k_{\mathrm{c}2}, k_{\mathrm{rG}})$ for the two extracts. We also fit a surface to the scattering of these points (translucent green surface plot), which helps visualize the fact that these points essentially lie on a two dimensional manifold within the three dimensional space of parameters, and that this surface displays thin covariation. The calibration step concludes with the projection of the points onto the environment specific parameter axes for $\mathcal{E}_1$ and $\mathcal{E}_2$, as shown by the filled-in blue circles on the horizontal $\log k_{\mathrm{c}}$–$\log[\mathrm{Enz}]$ planes in Figure 7D, E.

The red point in Figure 7D shows the result of the first correction step, where the environment specific parameters $([\mathrm{Enz}], k_{\mathrm{c}})$ were fixed to one of the points estimated in the calibration step, and the process specific parameter was estimated. We see that the process specific parameter value estimated is such that the full parameter point $(\hat{\theta}_{e2}, \hat{\theta}_p)$ lies in the joint environment- and process specific parameter set (red point lifted up to the green surface). The fitted trajectories from this stage are shown in Figure 7G.

We observe from the position of the red point in Figure 7E that picking an arbitrary point from the set of environment specific parameters, and using the process specific parameters from the first correction step leads to a point that does not lie on the joint environment- and process specific parameter surface for extract $\mathcal{E}_1$. The corresponding predicted trajectory and the true behavior of the artificial data are shown in Figure 7H, and do not match.

Figure 7F, I show the result of repeating the procedure with process specific parameter conditioning applied at the calibration step. In particular, the process specific parameter was fixed at the value that was estimated at the first calibration step (lifted red point in Figure 7D), though any value in the set $\mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}$ is allowed. The key insight here is that now the environment

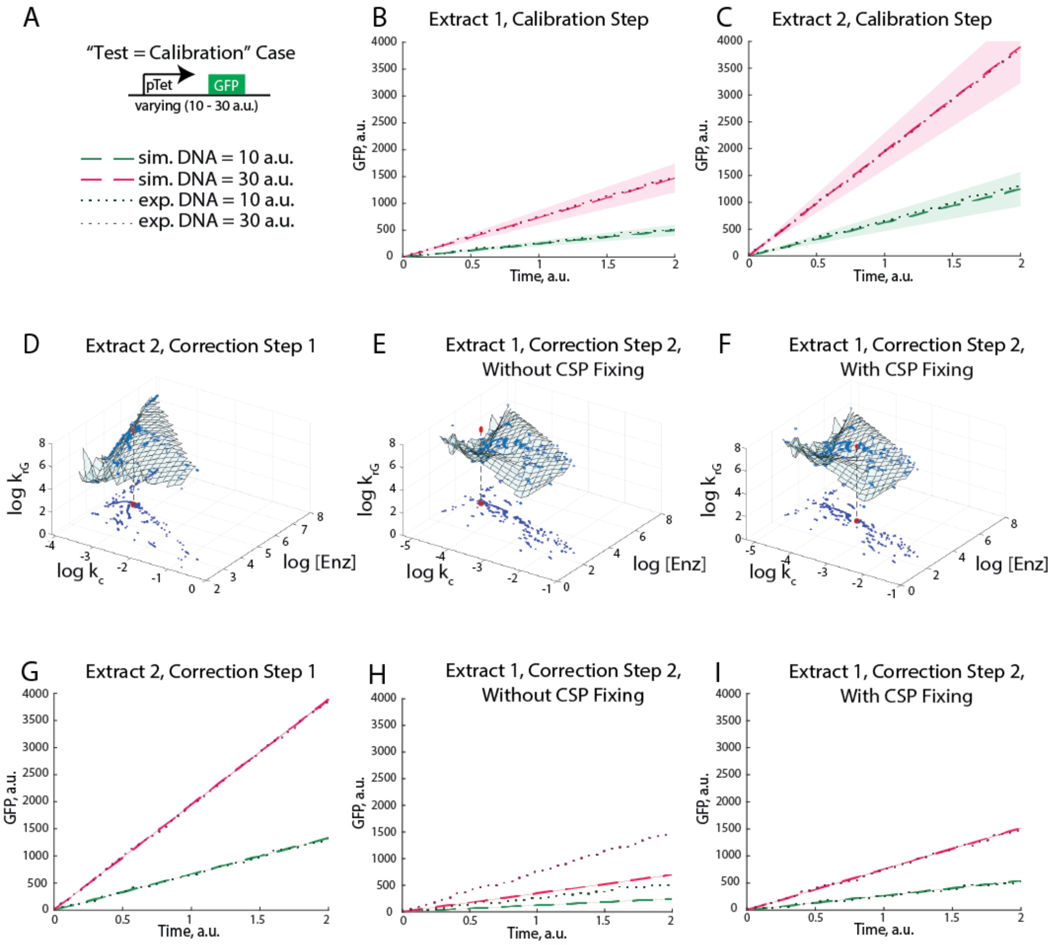

Figure 7: *In silico* demonstration of the effect of thin covariation and process specific parameter conditioning for the 'Test = Calibration' case. (A) Schematic showing the constitutive expression circuit (green fluorescent protein produced constantly). Legend shows what the lines in the subsequent panels mean. (B - C) Artificial data (dotted lines) generated using known fixed parameters for two extracts and the fitted trajectories (dashed lines) resulting from the calibration step. A small amount of noise was added to the dotted lines for easier visualization, since they overlapped almost perfectly with the fitted trajectories. See main text for details on how the data were generated. The dashed lines and shaded regions were the means and standard deviations of simulated trajectories using parameter points drawn from the estimated posterior distributions. (D, E) The blue points suspended in $\mathbb{R}^3$ are the parameter sets estimated at the calibration step, jointly between the two extracts (that is, in the 5 dimensional parameter space comprising one process specific parameter and two environment specific parameters per extract). The translucent green surfaces in both panels have been fitted to these points. They denote the set of all parameter points that fit extract $\mathcal{E}_1$ and extract $\mathcal{E}_2$ data to the model, and visually depict thin covariation in the process specific parameter ($\log k_{\mathrm{rG}}$) coordinates with respect to the environment specific parameter coordinates. Continued below.

Figure 7: Continued. The set of environment specific parameters estimated at the calibration stage are generated by projecting the points to the horizontal planes ($\log k_{\mathrm{c}}$–$\log[\mathrm{Enz}]$ planes in each panel), and are depicted as the dark blue points on these planes. (D) also shows the result of correction step 1, with the red point on the $\log k_{\mathrm{c}}$–$\log \mathrm{Enz}$ plane showing the arbitrary point $\hat{\theta}_{e2} \in E_{2,\mathrm{cal}}$ that was picked, and the corresponding red point lifted to the green surface showing the value of the estimated process specific parameter $\theta_p \equiv k_{\mathrm{rG}}$ that was estimated. (E) shows correction step 2 without process specific parameter conditioning, whereby combining an arbitrary point $\hat{\theta}_{e1} \in E_{1,\mathrm{cal}}$ with the process specific parameter $\hat{\theta}_p$ from (D) does not necessarily lead to a point $(\hat{\theta}_{e1}, \hat{\theta}_p)$ that lies on the green surface (red point). In (F), with process specific parameter conditioning, this situation is rectified, and the red point lies on the green surface. (G) The process model fits involved in correction step one. (H) The predicted trajectories lie away from the true trajectories because the red point in (E) does not lie on the green surface. (I) The predicted trajectories in correction step 2 with process specific parameter conditioning match the true trajectories.

specific parameter sets are much smaller, and due to the covariation being thin in the process specific parameter coordinates, in correction step one, the environment specific parameters can only be picked so that the very process specific parameter value that was fixed gets estimated. Subsequently, in correction step two, the only environment specific parameter values that can be picked are such that when they are used with this process specific parameter value, the resulting point lies in the set of parameters $\Theta_{1,\mathrm{cal}}$ that fit the true $\mathcal{E}_1$ data to the model. Indeed, in Figure 7I, we see that this leads to the desired correction.

8.2 Application of Process Specific Parameter Conditioning in the General Setting

We conclude this section by demonstrating that in the more general setting of the test circuit being different from the calibration circuit, process specific parameter conditioning can still help achieve significant improvements in the performance of the method (Figure 8). The calibration data used were the same as in Section 8.1, and the test circuit model (Figure 8A) used was the repression circuit, modeled by Equations (6), with parameters used to generate the artificial data given in Table **??** (Appendix C). As before, dotted curves denote artificial data with a small amount of noise added for ease of visualization. The calibration stage with and without process specific parameter conditioning was identical to that in Section 8.1. To reduce the dimension of the space that the parameter inference algorithm would need to explore, we fixed the forward rates k_{fG}, k_{fT},$k_{\mathrm{f,dim}}$, and $k_{\mathrm{f,rep}}$, and limited the process specific parameters to only the reverse rates, k_{rG}, k_{rT},$k_{\mathrm{r,dim}}$, and $k_{\mathrm{r,rep}}$. In this setting, performing the first correction step gave a set of parameter estimates for the process specific parameters, and the resulting fits to the $\mathcal{E}_2$ test circuit data are shown in Figure 8B. Performing the second correction step without process specific parameter conditioning led the expected incorrect prediction of the corrected trajectories (Failure Condition 2), as shown in Figure 8C. Finally, applying process specific parameter conditioning to the calibration step led to good prediction of the circuit behaviour at correction step 2, as shown in Figure 8D.

9 Summary and Discussion

Calibrating and correcting environment specific effects is an important problem in disciplines as varied as aeronautics and synthetic biology. Model-based methodologies can be used in conjunction with calibration experiments to learn environment specific parameters, which in turn can be used to correct for differences in a system's behavior across environments [1].

The models used in these methodologies can often possess parameters that cannot be uniquely determined given the observable data (initial conditions, inputs and output trajectories), even in the absence of observation noise. This property is called *structural* non-identifiability in the control literature, and is dependent solely on the structure of a model's equations [13]. The presence of structural non-identifiability often limits the use of models in applications such as system composition or variability reduction. We have studied the interaction of this phenomenon with a general calibration methodology, and derived necessary and sufficient conditions under

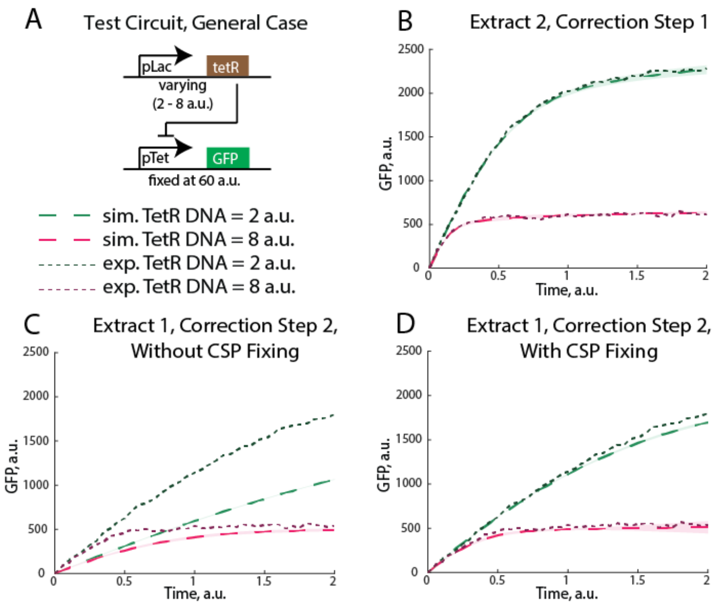

Figure 8: The effect of process specific parameter conditioning on the correction of novel test circuit data (that is, of data not seen at the calibration stage). (A) The test circuit was the repression of the pTet promoter, modeled by Equations (6). The pTet-GFP DNA was held fixed at 60 a.u., while the constitutive TetR DNA was varied between 2 a.u. and 8 a.u.. The dotted curves were the artificial experimental data generated using the parameters in Table **??**. The calibration step was performed as in Figure 7, both with and without process specific parameter conditioning. The first correction step leads to the fits shown in (B), and the second correction step leads to the poor corrections shown in (C). When process specific parameter conditioning is employed at the calibration step, the second correction step performs well, as shown in (D).

which it does not hinder such methodologies. In the following subsections, we review some key aspects of our results, and discuss noteworthy generalizations and future directions.

9.1 Correcting Genetic Circuit Behavior Across Extracts

To experimentally demonstrate the calibration methodology and show the implications of the main theoretical results, we chose an application from the field of synthetic biology, where models of genetic 'circuits' (networks) tend to have a large number of non-identifiable parameters. In this field, cell-free 'extracts' are becoming a useful environments for prototyping such genetic circuits. Indeed, the intrinsic variability between the batches of extracts places limitations on our ability to compare results from different batches. Users currently plan their investigations so that their experiments may be completed before the batch of extract is depleted, and are therefore limited in the number of results they are able to compare under identical experimental conditions.

The methodology is organized into two steps, a calibration step, where a set of calibration circuits is used to estimate extract (environment) specific parameters of a particular extract, and a correction step, in which the calibrations are used to tranform a novel circuit's behavior from what it was in a given extract into what it would have been in a *reference* extract. The idea is that whenever a new extract batch is made, a predefined set of calibration experiments may be performed on that extract to measure its environment specific parameters. These, along with similarly estimated parameters for the reference extract may be used to transform any data collected in the new extract into the reference extract form, and thus be made directly comparable with all other data also similarly transformed.

9.2 Correcting Across Topologically Distinct Environments

We have developed this method for normalizing behavior across extract batches that are assumed to differ only in the values of the parameters of the reaction network for a given circuit, and not in the (graph) *topology* of the networks modeling the extracts. The framework here should be applicable to any scenario where only such parametric differences exist. One example of this situation is when correcting for run-to-run variability between experiments, which would require calibration experiments to be performed alongside each run.

The more general situation of correcting data between topologically different environments can occur when predicting circuit behavior in cells (*in-vivo*) from measurements in extracts (*in-vitro*), or when correcting for variability between different bacterial strains or microorganism species. In this more general setting, as long as the modeling framework and environment specific parameters are chosen carefully enough to capture most of the environment specific influences on the circuit, then the process specific parameters should be largely independent of the environment they are estimated in. The appropriateness of the choice of the level of detail in the modeling, the partition of parameters into environment specific versus process specific and the choice of calibration experiments in each of the different environments may be achieved in an iterative, empirical, and hypothesis driven manner.

9.3 Parameter Consistency Conditions

We have also developed theoretical results for when this methodology is expected to work in the presence of parameter non-identifiability. Due to the large discrepancy between the size of biochemical networks and the number of molecular species that can be measured, parameter non-identifiability is a ubiquitous property of these reaction network models. The general prescription in modeling studies [16] is to perform more experiments to eliminate non-identifiability, reduce the order of the model by lumping parameters and equations, or to fix some parameters to effectively reduce the number of non-identifiable parameters. However, in many cases, more experiments may not be feasible due to cost, time or technological constraints. Model order reduction may not be desirable if, for example, certain mechanisms in the model need to be kept separate (one example being the explicit modeling of nucleotide binding and consumption during transcription and translation to keep track of resources [14]). The fixing of some parameters, while reducing the number of effective parameters may not remove non-identifiability completely.

The key point behind our results is that since we are trying to correct the trajectories of the very species that we are able to measure, the sets of values the non-identifiable parameters can take could perhaps be treated as equivalence classes with respect to their usage in the modeling framework. This is a general idea that, even though developed and demonstrated in this specific calibration framework, should apply to a broader class of applications of parametric models, as long as those applications depend on using only the observable outputs. One future direction of this work would be to develop these ideas at this level of generality, starting with the linear systems framework found in control theory.

9.4 Other Future Directions

We may identify a few other directions of investigation. First, it might be possible to generalize condition 15 in Theorem 1 to a result which gives conditions under which part models with non-identifiability can be combined to predict the behavior of an entire system. That is, to derive conditions under which models can be composed into larger models in a way that the combined model is still predictive of real behavior, despite structural non-identifiability in the part models. In the simplest case, this could be a simple Cartesian product condition, though it is likely that this would be too restrictive, since covariation between the parameters of different parts may exist, requiring a more careful analysis. For example, we may have to prescribe precisely which parameters must be identified, and to what extent, and which parameters may be fixed (analogous to process specific parameter conditioning above), before the remaining non-identifiability does not hinder the model's ability to predict system behavior.

Second, it has been noted in the control and systems biology literature that the sets of output-indistinguishable parameters find their most natural description as differentiable manifolds [17, 18]. These manifolds foliate the parameter space, with the individual leaves of the foliation corresponding to different sets of outputs (for a given set of inputs and initial conditions). Classical nonlinear control [19, 20, 21] provides a rich theory for understanding these manifolds, and it should be possible to use this framework characterize the relationship between the geometric structure of these manifolds and the different choices of inputs, output variables, and initial conditions. The understanding of such relationships could be useful for both the variability reduction problem (this work) or the more general system composition problem: beginning with the parameter consistency conditions, we might be able to *design* experiments (that is, decide on which outputs to measure, what initial conditions to test, and what inputs to apply) that reduce the non-identifiability so that the consistency conditions are met.

Finally, we may wish to generalize these results to the case when there is noise in the data, the parameter sets are replaced by probability distributions, and notions of practical identifiability [22] are incorporated into our analysis.

Acknowledgments and Disclosure of Funding

This work was supported by SBIR-STTR grant W911NF-16-P-0003 and AFOSR grant FA9550-14-1-0060. The authors would like to thank Samuel Clamons, Wolfgang Halter, William Poole, Anandh Swaminathan and Andras Gyorgy for useful discussions, and James Parkin, Shaobin Guo, Mark Prator, Georgios Artavanis and Abel Chiao for providing additional batches of extracts to test.

A Proofs of Main Results

In this section, we provide detailed proofs of the main results of this study.

Proof of Theorem 1. Solving the data correction problem using the calibration-correction method involves avoiding failure conditions one and two described in Definitions 8 and 9. Avoiding failure condition one wherever it may occur ensures that the method can be implemented in the first place, and avoiding failure condition two means that the method returns the desired result.

The necessity of condition (13) follows from the fact that if $\tilde{\Theta}_{\text{cal}} = \emptyset$, then there does not exist a vector $(\theta_{e1}, \theta_{e2}, \theta_p)$ such that $\overline{y}_{i,\text{cal}} = M_{\text{cal}}(\theta_{e,i}, \theta_p)$ for $i = 1, 2$, leading to failure condition one at the calibration step. We note that in the model universe, where $M_{\text{cal}}(\theta) = \overline{M}_{\text{cal}}(\overline{\theta})$ and $\overline{y}_{i,\text{cal}} = \overline{M}_{\text{cal}}(\overline{\theta_e}, \overline{\theta_p})$, condition (13) always holds.

Next, we prove the necessity of $E_{2,\text{cal}} \subseteq E_{2,\text{test}}$, where $E_{2,\text{test}} \triangleq \operatorname{proj}_{\theta_e} \Theta_{2,\text{test}}$. Assume that there exists an $\tilde{\theta}_e \in E_{2,\text{cal}}$ such that $\tilde{\theta}_e \notin E_{2,\text{test}}$. Thus, there does not exist a $\tilde{\theta}_p$ such that $M_{\text{test}}((\tilde{\theta}_e, \tilde{\theta}_p)) = \overline{y}_{2,\text{test}}$. Since the operator $\text{ID}_{\theta_p|\theta_e=\tilde{\theta}_e}$ is only defined on the set $\{(y, M) \mid \exists \theta_p : M((\tilde{\theta}_e, \theta_p)) = y\}$, we see that the map $\text{ID}_{\theta_p|\theta_e=\tilde{\theta}_e}(\overline{y}_{2,\text{test}}, M_{\text{test}}(\theta_e, \theta_p))$ is not well defined, leading to failure condition one at the first correction step.

We prove the necessity of condition (15) as follows. Assume that there exists an $(\tilde{\theta}_e, \tilde{\theta}_p) \in E_{1,\text{cal}} \times P'_{2,\text{test}}$ such that $(\tilde{\theta}_e, \tilde{\theta}_p) \notin \Theta_{1,\text{test}}$. Since we use points $\hat{\theta}_e \in E_{1,\text{cal}}$ and $\hat{\theta}_p \in P'_{2,\text{test}}$ to generate the prediction $\hat{y}_{1,\text{test}}$ in the second correction step, it is possible that $\hat{\theta}_e = \tilde{\theta}_e$ and $\hat{\theta}_p = \tilde{\theta}_p$. Furthermore, since $\Theta_{1,\text{test}}$ is the set of all points (θ_e, θ_p) that give the correct trajectory $\overline{y}_{1,\text{test}}$, we have the possibility that $\hat{y}_{1,\text{test}} \neq \overline{y}_{1,\text{test}}$. This is the second failure condition.

Finally, sufficiency is a simple consequence of the fact that conditions (13-15) address both the points in the method where failure condition one could be met, and the point in the method where failure condition two could occur. Explicitly, condition (13) allows the calibration step to avoid failure condition one, condition (14) allows correction step one to avoid failure condition one, since it implies that for all $\tilde{\theta}_e \in E_{2,\text{cal}}$, there exists a $\tilde{\theta}_p$ such that $(\tilde{\theta}_e, \tilde{\theta}_p) \in \Theta_{2,\text{test}}$. Condition (15) enables correction step two to avoid failure condition two, since it implies that for all $\tilde{\theta}_e \in E_{1,\text{cal}}$ and for all $\tilde{\theta}_p \in P'_{2,\text{test}}$ we have that $\overline{y}_{1,\text{test}} = M_{\text{test}}(\tilde{\theta}_e, \tilde{\theta}_p)$, implying that the set of all possible predicted trajectories only has the correct trajectory in it, $Y_1 = \{\overline{y}_{1,\text{test}}\}$. □

Proof of Corollary 1. Recall from Remark 1 that in the model universe, the data are generated by nominal parameters, $\overline{\theta}_{e1}$, $\overline{\theta}_{e2}$, $\overline{\theta}_{p,\text{cal}}$, $\overline{\theta}_{p,\text{test}}$. We observe that since the models are structurally globally identifiable, these parameters uniquely fit the model to the data, and therefore the sets in conditions (13-15) only have single entries, leading to these conditions being trivially satisfied:

$$\begin{aligned}
&\tilde{\Theta}_{\text{cal}} = \{(\overline{\theta}_{e1}, \overline{\theta}_{e2}, \overline{\theta}_{p,\text{cal}})\} \neq \emptyset, \\
&E_{2,\text{cal}} = \{\overline{\theta}_{e2}\} \subseteq \operatorname{proj}_{\theta_e} \{(\overline{\theta}_{e2}, \overline{\theta}_{p,\text{test}})\} = \operatorname{proj}_{\theta_e} \Theta_{2,\text{test}}, \\
&E_{1,\text{cal}} \times P'_{2,\text{test}} = \{\overline{\theta}_{e1}\} \times \{\overline{\theta}_{p,\text{test}}\} \subseteq \{(\overline{\theta}_{e1}, \overline{\theta}_{p,\text{test}})\} = \Theta_{1,\text{test}}.
\end{aligned}$$

□

Proof of Lemma 1. First, we prove the 'only if' direction. Covariation implies that there exists a point $\tilde{\theta}_b \in \operatorname{proj}_{\theta_b} \Theta$ such that

$$\tilde{\theta}_b \in \big(\operatorname{proj}_{\theta_b} \big(\Theta \cap \operatorname{cut}_{\theta_b}(\tilde{\theta}_{a1})\big) \triangle \big(\operatorname{proj}_{\theta_b} \big(\Theta \cap \operatorname{cut}_{\theta_b}(\tilde{\theta}_{a2})\big),$$

where $\triangle$ is the symmetric difference set operation. It further implies that there exists a point $\tilde{\theta}_a \in \{\tilde{\theta}_{a1}, \tilde{\theta}_{a2}\} \in \operatorname{proj}_{\theta_a} \Theta$ such that $(\tilde{\theta}_a, \tilde{\theta}_b) \notin \Theta$. Thus, $\operatorname{proj}_{\theta_a} \Theta \times \operatorname{proj}_{\theta_b} \Theta \neq \Theta$.

Next, we prove the 'if' direction. Let $(\tilde{\theta}_a, \tilde{\theta}_b) \in \operatorname{proj}_{\theta_a} \Theta \times \operatorname{proj}_{\theta_b} \Theta$ be such that $(\tilde{\theta}_a, \tilde{\theta}_b) \notin \Theta$. Since $\tilde{\theta}_b \in \operatorname{proj}_{\theta_b} \Theta$, there exists a $\tilde{\theta}_{a2} \in \operatorname{proj}_{\theta_a} \Theta$ such that $(\tilde{\theta}_{a2}, \tilde{\theta}_b) \in \Theta$. Thus we have $\tilde{\theta}_b \in \operatorname{proj}_{\theta_b} \big(\Theta \cap \operatorname{cut}_{\theta_b}(\tilde{\theta}_{a2})\big)$ but $\tilde{\theta}_b \notin \operatorname{proj}_{\theta_b} \big(\Theta \cap \operatorname{cut}_{\theta_b}(\tilde{\theta}_{a1})\big)$, which proves the assertion. □

Proof of Proposition 1. Condition (22) and the fact that for the 'Test = Calibration' case, $P'_{2,\mathrm{cal}} = \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}$, together imply that $\mathrm{proj}_{\theta_p} \Theta'_{1,\mathrm{cal}} = P'_{2,\mathrm{cal}}$. Condition (23) implies $\mathrm{proj}_{\theta_e} \Theta'_{1,\mathrm{cal}} = E_{1,\mathrm{cal}}$. Covariation implies that $\mathrm{proj}_{\theta_e} \Theta'_{1,\mathrm{cal}} \times \mathrm{proj}_{\theta_p} \Theta'_{1,\mathrm{cal}} \neq \Theta'_{1,\mathrm{cal}}$. Thus, the proper subset relation $\Theta'_{1,\mathrm{cal}} \subsetneq E_{1,\mathrm{cal}} \times P'_{2,\mathrm{cal}}$ holds, and therefore there exists $(\tilde{\theta}_e, \tilde{\theta}_p) \in E_{1,\mathrm{cal}} \times P'_{2,\mathrm{cal}}$ such that $(\tilde{\theta}_e, \tilde{\theta}_p) \notin \Theta'_{1,\mathrm{cal}} \subseteq \Theta_{1,\mathrm{cal}}$. This implies that $E_{1,\mathrm{cal}} \times P'_{2,\mathrm{cal}} \nsubseteq \Theta_{1,\mathrm{cal}}$, which violates condition (20). □

Proof of Proposition 2. Let $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}$ and $\tilde{\theta}_{e2} \in E_{2,\mathrm{cal}} \triangleq \mathrm{proj}_{\theta_e} \left(\mathrm{cut}_{\theta_e} \left(\tilde{\theta}_p\right) \cap \Theta_{2,\mathrm{cal}}\right)$. We note that the sets $\mathrm{proj}_{\theta_p} \left(\mathrm{cut}_{\theta_p} \left(\tilde{\theta}_{e2}\right) \cap \Theta_{2,\mathrm{cal}}\right)$ and $\mathrm{ID}_{\theta_p}(\overline{y}_{2,\mathrm{cal}}, M_{\mathrm{cal}}\left(\tilde{\theta}_{e2}, \theta_p\right))$ are equal by definition. Now, pick an arbitrary point $\tilde{\theta}_p{}' \in \mathrm{proj}_{\theta_p} \left(\mathrm{cut}_{\theta_p} \left(\tilde{\theta}_{e2}\right) \cap \Theta_{2,\mathrm{cal}}\right)$. It follows that $\tilde{\theta}_p{}' = \tilde{\theta}_p$ from the fact that $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \left(\mathrm{cut}_{\theta_p} \left(\tilde{\theta}_{e2}\right) \cap \Theta_{2,\mathrm{cal}}\right)$ and that the element in $\left|\mathrm{cut}_{\theta_a}(\tilde{\theta}_b) \cap \Theta\right| = 1$ is unique. Thus, the only possible process specific parameter value that can be returned by the first correction step is $\tilde{\theta}_p$.

Next, we look at the second correction step. Pick an arbitrary $\tilde{\theta}_{e1}$ in $E_{1,\mathrm{cal}}$ and recall that $E_{1,\mathrm{cal}} \triangleq \mathrm{proj}_{\theta_e} \left(\mathrm{cut}_{\theta_e} \left(\tilde{\theta}_p\right) \cap \Theta_{1,\mathrm{cal}}\right)$. Since the point $\left(\tilde{\theta}_{e1}, \tilde{\theta}_p\right) \in \Theta_{1,\mathrm{cal}}$, we have that $\overline{y}_{1,\mathrm{cal}} = \hat{y}_{1,\mathrm{cal}} \triangleq M(\tilde{\theta}_{e1}, \tilde{\theta}_p)$, and failure condition two is avoided. □

B Supplementary Lemmas

In this section, we give further results that help clarify some issues mentioned in the main text, and provide details on the experimental and computational methodology.

B.1 Equivalence of the Two Definitions of the Calibration Step

In this section, we prove two identities that establish the equivalence of the two definitions of the calibration step given in Definition 7 and the directly following it.

Lemma 2. *Let* $\tilde{\Theta}_{\mathrm{cal}}$*,* $\Theta_{1,\mathrm{cal}}$ *and* $\Theta_{2,\mathrm{cal}}$ *be as defined in Definition 7 and the text following it. Then, the identities*

$$\mathrm{proj}_{\theta_p} \tilde{\Theta}_{\mathrm{cal}} \equiv \mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}, \tag{24}$$

$$\mathrm{proj}_{\theta_{e,i}} \tilde{\Theta}_{\mathrm{cal}} \equiv \left\{ \theta_e \,\middle|\, \exists \theta_p \in \left(\mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}\right) : (\theta_e, \theta_p) \in \Theta_{i,\mathrm{cal}}\right\}, \qquad i = 1, 2, \tag{25}$$

hold.

Proof. First, we prove (24) using a series of equivalences. Let $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \tilde{\Theta}_{\mathrm{cal}}$. This is equivalent to

$$\exists \theta_{e1}, \theta_{e2} : (\theta_{e1}, \theta_{e2}, \tilde{\theta}_p) \in \tilde{\Theta}_{\mathrm{cal}} \tag{26}$$

$$\Leftrightarrow \exists \theta_{e1}, \theta_{e2} : \overline{y}_{i,\mathrm{cal}} = M_{\mathrm{cal}}(\theta_{e,i}, \tilde{\theta}_p), \qquad i = 1,2 \tag{27}$$

$$\Leftrightarrow (\theta_{e,i}, \tilde{\theta}_p) \in \Theta_{i,\mathrm{cal}}, \qquad i = 1,2 \tag{28}$$

$$\Leftrightarrow \tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}, \tag{29}$$

which proves the assertion.

Next, we prove (25) for θ_{e1} by showing that the left and right hand sides are subsets of each other. The proof for the θ_{e2} case is similar. Denote the set on the left hand side with L, and the one on the right with R. Let $\tilde{\theta}_{e1} \in L = \mathrm{proj}_{\theta_{e1}} \tilde{\Theta}_{\mathrm{cal}}$. Then, $\exists \tilde{\theta}_{e2}, \tilde{\theta}_p$ such that $(\tilde{\theta}_{e1}, \tilde{\theta}_{e2}, \tilde{\theta}_p) \in \tilde{\Theta}_{\mathrm{cal}}$, which implies $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \tilde{\Theta}_{\mathrm{cal}}$ and $\overline{y}_{1,\mathrm{cal}} = M_{\mathrm{cal}}(\tilde{\theta}_{e1}, \tilde{\theta}_p)$. By the identity (24), we have that $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}$ and $(\tilde{\theta}_{e1}, \tilde{\theta}_p) \in \Theta_{1,\mathrm{cal}}$, which shows that $L \subseteq R$.

We conclude the proof by showing that $R \subseteq L$. Let $\tilde{\theta}_{e1} \in R$, which means that there exists a $\tilde{\theta}_p \in \mathrm{proj}_{\theta_p} \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_{\theta_p} \Theta_{2,\mathrm{cal}}$ such that $\overline{y}_{1,\mathrm{cal}} = M_{\mathrm{cal}}(\tilde{\theta}_{e1}, \tilde{\theta}_p)$. Furthermore, since $\tilde{\theta}_p \in$

$\text{proj}_{\theta_p}\Theta_{2,\text{cal}}$, there also exists an $\tilde{\theta}_{e2}$ such that $\overline{y}_{2,\text{cal}} = M_{\text{cal}}(\tilde{\theta}_{e2}, \tilde{\theta}_p)$. Together these imply that $(\tilde{\theta}_{e1}, \tilde{\theta}_{e2}, \tilde{\theta}_p) \in \tilde{\Theta}_{\text{cal}}$, which gives $\tilde{\theta}_{e1} \in \text{proj}_{\theta_{e1}} \tilde{\Theta}_{\text{cal}}$, proving the assertion. □

B.2 Equivalence of the Two Process Specific Parameter Subset Conditions Given in the Text following Theorem 1

The Cartesian product condition given in equation (15) implies two further conditions, which we state in Lemma 3 below. The first of these follows simply by projecting both sides of equation (15) onto the environment specific parameter coordinates. The second condition, on the other hand, is stronger than simply projecting (15) onto the process specific parameter coordinates. This condition states that the process specific parameter points generated at the first correction step, $P'_{2,\text{test}}$, must be a subset of the set of process specific parameter points generated by fitting $\overline{y}_{1,\text{test}}$ to the model when the environment specific parameter points are restricted to be in the set $E_{1,\text{cal}}$.

Lemma 3. *Condition* (15)*, which states that* $E_{1,\text{cal}} \times P'_{2,\text{test}} \subseteq \Theta_{1,\text{test}}$*, implies that*

$$E_{1,\text{cal}} \subseteq \text{proj}_{\theta_e} \Theta_{1,\text{test}}, \tag{30}$$

$$P'_{2,\text{test}} \subseteq P'_{1,\text{test}}, \tag{31}$$

where $P'_{1,\text{test}}$ *is defined in a similar way to* $P'_{2,\text{test}}$*,*

$$P'_{1,\text{test}} \triangleq \bigcup_{\hat{\theta}_e \in E_{1,\text{cal}}} \text{ID}_{\theta_p|\theta_e=\hat{\theta}_e}\left(\overline{y}_{1,\text{test}}, M_{\text{test}}\big(\theta_e, \theta_p\big)\right).$$

Proof. Condition (30) follows simply by applying the proj_{θ_e} operator to both sides of condition (15). To prove condition (31), we note that condition (15) implies that for an arbitrary $\tilde{\theta}_p \in P'_{2,\text{test}}$, we have that for all $\tilde{\theta}_e \in E_{1,\text{cal}}$, the model fits the data, $\overline{y}_{1,\text{test}} = M_{\text{test}}(\tilde{\theta}_e, \tilde{\theta}_p)$. This in turn implies that

$$\tilde{\theta}_p \in \bigcup_{\hat{\theta}_e \in E_{1,\text{cal}}} \text{ID}_{\theta_p|\theta_e=\hat{\theta}_e}\left(\overline{y}_{1,\text{test}}, M_{\text{test}}\big(\theta_e, \theta_p\big)\right) = P'_{1,\text{test}}. \tag{32}$$

Thus, $P'_{2,\text{test}} \subseteq P'_{1,\text{test}}$. □

C Methods

C.1 TX-TL Extract and Buffer Preparation

Preparation and execution of TX-TL was according to previously described protocols [23].

Briefly, the cells were grown to an OD_{600} of 1.5, pelleted and washed. They were then lysed using a french press, and centrifuged to remove cell debris. The supernatant was incubated at 37°C for 80 min, and then centrifuged to remove endogenous nucleic acids. The supernatant was dialyzed against a pH8.2 buffer containing Mg-glutamate, K-glutamate, Tris, and DTT. Finally, the extract was centrifuged and the supernatant was flash-frozen in liquid nitrogen and stored at -80°C.

The buffer had the following components: 9.9 mg/mL protein, 9.5 mM Mg-glutamate, 95 mM K-glutamate, 0.33 mM DTT, 1.5 mM each amino acid except leucine, 1.25 mM leucine, 50 mM HEPES, 1.5 mM ATP and GTP, 0.9 mM CTP and UTP, 0.2 mg/mL tRNA, 0.26 mM CoA, 0.33 mM NAD, 0.75 mM cAMP, 0.068 mM folinic acid, 1 mM spermidine, 30 mM 3-PGA, 2% PEG-8000. Both the extract and buffer were stored at -80°C in separate tubes, with enough volume for seven reactions per tube.

C.2 Cell Extract Experiment

A 384-well microplate (Nunc) was used for the experiments, and the appropriate concentrations and volumes of DNA and inducers to be used in each reaction were calculated using the spreadsheets provided in [23]. The extract and buffer were thawed for 20 min on ice, mixed in the prescribed ratios, and pipetted into each well being used in the microplate, which was also placed on ice. The DNA was then added to each well according to the spreadsheet using an Echo 525 liquid handler robot. All the pipetting was done to avoid bubbles, the plate was sealed, and spun at 4000g for 45s at 4°C to distribute the mix evenly at the bottom of the wells and remove any bubbles that might have been introduced. The plate was placed in a Synergy H1/MF microplate reader (Biotek). Settings used for deGFP measurement were: excitation/emission 485 nm/515 nm, at gain 61, measured every 8 min for 8 hours.

D Additional Details Associated with the Models in Section ??

D.1 Calibration Circuit

D.2 The Test Circuit

Then, the first correction step involves inferring the distribution of $\theta_{p,\text{test}} = (k_{\text{fG}}, k_{\text{rG}}, (k_{\text{fT}}, k_{\text{rT}}, k_{\text{f,rep}}, k_{\text{r,rep}}, k_{\text{f,dim}}, k_{\text{r,dim}})$ from the set of data-model pairs in the candidate extract $\mathcal{E}_2$, over the different initial conditions,

$$\left\{ \begin{matrix} \left(\overline{y}_{2,\text{test},1}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,1}\right)\Big), & \left(\overline{y}_{2,\text{test},2}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,2}\right)\Big), \\ \left(\overline{y}_{2,\text{test},3}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,3}\right)\Big), & \left(\overline{y}_{2,\text{test},4}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,4}\right)\Big), \\ \left(\overline{y}_{2,\text{test},5}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,5}\right)\Big), & \left(\overline{y}_{2,\text{test},6}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,6}\right)\Big), \\ \left(\overline{y}_{2,\text{test},7}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,7}\right)\Big), & \left(\overline{y}_{2,\text{test},8}\right), M_{\text{test}}\left((\hat{\theta}_{e2,\text{cal}}, \theta_{p,\text{test}}), x_{0,2,8}\right)\Big) \end{matrix} \right\}. \tag{33}$$

The second correction step involves predicting the the behavior of the system in the reference extract $\mathcal{E}_1$ at all eight initial conditions. Since a distribution for $\theta_{p,\text{test}}$ is generated at correction step 1, we sample 500 points from this distribution, and plot the mean and standard deviation of the predictions due to these samples,

$$\left\{ \begin{matrix} \left(\hat{y}_{1,\text{test},1}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,1}\right)\Big), & \left(\hat{y}_{1,\text{test},2}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,2}\right)\Big), \\ \left(\hat{y}_{1,\text{test},3}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,3}\right)\Big), & \left(\hat{y}_{1,\text{test},4}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,4}\right)\Big), \\ \left(\hat{y}_{1,\text{test},5}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,5}\right)\Big), & \left(\hat{y}_{1,\text{test},6}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,6}\right)\Big), \\ \left(\hat{y}_{1,\text{test},7}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,7}\right)\Big), & \left(\hat{y}_{1,\text{test},8}\right), M_{\text{test}}\left((\hat{\theta}_{e1,\text{cal}}, \hat{\theta}_{p,\text{test}}), x_{0,1,8}\right)\Big) \end{matrix} \right\}. \tag{34}$$

Note that even though the fitting and the prediction in correction steps 1 and 2 were done on all eight values of D_{T0}, we only show the results for the first four of these conditions in Figure 4C. The remaining trajectories are very close to 0, being repressed more strongly than even the 0.75 nM trajectory in the bottom row of Figure 4C, and are suppressed for brevity.

E Key Synthetic Biology Terms

In this section, we provide a brief primer of concepts from synthetic and molecular biology that are useful for understanding the examples described in this work.

- *Central Dogma*: Describes the flow of genetic information within cells, and states that DNA is used to produce RNA (transcription), which is then used to produce proteins (translation).
- *Promoter*: Regions of DNA located upstream of a the gene sequence, which encodes proteins. Being upstream, these regions are accessed before the gene during transcription, and regulating this access can regulate whether the protein is expressed. Specifically, promoters serve as docking sites for RNA polymerase, the enzyme responsible for transcription, and for transcription factors, which are proteins that bind to them and can regulate the recruitment of RNA polymerase for transcription.
- *Transcription Factor*: Transcription factors are proteins that bind to specific DNA sequences within promoters. They act as molecular switches that activate or repress gene expression. They do this by influencing the recruitment of RNA polymerase and other components of the transcriptional machinery to the promoter, thereby controlling the rate of transcription. By engineering the interactions between transcription factors and

promoters, synthetic biologists can design precise regulatory circuits. The repressor discussed in this study is the transcription factor TetR, repressing the promoter pTet.

- *Plasmid*: Small, circular DNA molecules that exist separate from the chromosomal DNA in many bacteria and other organisms. They are commonly used as carriers for the introduction of foreign genes into cells. Creating and introducing plasmids into living cells is a time consuming process, making the design of genetic circuits difficult, and motivating the use of cell extracts as a prototyping platform.

- *Cloning*: A process in which a desired gene or DNA sequence is inserted into a plasmid using restriction enzymes and ligases (also a type of enzymes). The newly created ‘recombinant’ plasmid can then be introduced into a host cell, where it replicates and expresses the inserted gene or set of genes.

- *Genetic circuit*: Engineered networks of genes and regulatory elements (such as promoters) that allow cells to perform specific tasks. Genetic circuits are composed of the so-called ‘transcriptional units’, which are essentially modules comprising a promoter, a gene (also known as a coding sequence), and other regulatory elements. Because the coding sequence can encode transcription factor proteins, which in turn can regulate the promoters in other transcriptional units, transcriptional units can be composed in a modular fashion to create complex circuits.

Traditionally, designing and testing genetic circuits required cloning genetic circuits onto plasmids. This is costly and time consuming for several reasons. First, cloning genetic circuits often involves the assembly of DNA fragments using techniques such as restriction enzyme digestion, ligation, and polymerase chain reaction (PCR). These processes can be labor-intensive and error prone. Furthermore, once the genetic circuit is cloned into a plasmid, it needs to be introduced into a host organism, typically bacteria. This process can also fail, leading to a failed cloning attempt. For each iteration of genetic circuit design, this entire process must be repeated, making the design of genetic circuits using plasmids a prohibitively expensive and time consuming process.

Cell extracts, also known as cell-free systems, are an alternative that allow the iterative design of genetic circuits without the need to assemble plasmids and express them within cells. They allow the use of linear DNA fragments that can be assembled without the need for cloning, and introduced into extracts with ease, allowing for the genetic circuit encoded by the linear DNA fragments to be ‘implemented’. The concentrations of the linear DNA fragments added to the extract can be finely controlled, allowing for fine control on various parts of the circuit, a feature that their plasmid counterparts introduced into living cells do not possess. A fuller discussion of these considerations can be found in the literature [9, 24, 12].

References

[1] Robert S. Swanson and Clarence L. Gillis. Wind-Tunnel Calibration and Correction Procedures for Three-Dimensional Models. *NACA Wartime Reports*, L4E31, October 1944. URL http://ntrs.nasa.gov/search.jsp?R=19930090922.

[2] Sándor Vajda. Structural equivalence of linear systems and compartmental models. *Mathematical Biosciences*, 55(1):39–64, July 1981. ISSN 0025-5564. doi: 10.1016/0025-5564(81)90012-2. URL http://www.sciencedirect.com/science/article/pii/0025556481900122.

[3] Michael B. Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338, January 2000. ISSN 0028-0836. doi: 10.1038/35002125. URL http://www.nature.com/nature/journal/v403/n6767/abs/403335a0.html.

[4] Timothy S. Gardner, Charles R. Cantor, and James J. Collins. Construction of a genetic toggle switch in Escherichia coli. *Nature*, 403(6767):339–342, January 2000. ISSN 0028-0836. doi: 10.1038/35002131. URL http://www.nature.com/nature/journal/v403/n6767/full/403339a0.html.

[5] S. Mangan and U. Alon. Structure and function of the feed-forward loop network motif. *Proceedings of the National Academy of Sciences*, 100(21):11980–11985, October 2003. doi: 10.1073/pnas.2133841100. URL https://www.pnas.org/doi/10.1073/pnas.2133841100. Publisher: Proceedings of the National Academy of Sciences.

[6] Xu Yan, Xu Liu, Cuihuan Zhao, and Guo-Qiang Chen. Applications of synthetic biology in medical and pharmaceutical fields. *Signal Transduction and Targeted Therapy*, 8(1):1–33, May 2023. ISSN 2059-3635. doi: 10.1038/s41392-023-01440-5. URL https://www.nature.com/articles/s41392-023-01440-5. Number: 1 Publisher: Nature Publishing Group.

[7] Adam M. Beitz, Conrad G. Oakes, and Kate E. Galloway. Synthetic gene circuits as tools for drug discovery. *Trends in Biotechnology*, 40(2):210–225, February 2022. ISSN 0167-7799, 1879-3096. doi: 10.1016/j.tibtech.2021.06.007. URL https://www.cell.com/trends/biotechnology/abstract/S0167-7799(21)00137-2. Publisher: Elsevier.

[8] Behide Saltepe, Ebru Şahin Kehribar, Side Selin Su Yirmibeşoğlu, and Urartu Özgür Şafak Şeker. Cellular Biosensors with Engineered Genetic Circuits. *ACS Sensors*, 3(1):13–26, January 2018. doi: 10.1021/acssensors.7b00728. URL https://doi.org/10.1021/acssensors.7b00728. Publisher: American Chemical Society.

[9] Henrike Niederholtmeyer, Zachary Z. Sun, Yutaka Hori, Enoch Yeung, Amanda Verpoorte, Richard M. Murray, and Sebastian J. Maerkl. Rapid cell-free forward engineering of novel genetic ring oscillators. *eLife*, 4:e09771, October 2015. ISSN 2050-084X. doi: 10.7554/eLife.09771. URL https://elifesciences.org/articles/09771.

[10] Chelsea Y. Hu, Jeffrey D. Varner, and Julius B. Lucks. Generating Effective Models and Parameters for RNA Genetic Circuits. *ACS Synthetic Biology*, June 2015. doi: 10.1021/acssynbio.5b00077. URL http://dx.doi.org/10.1021/acssynbio.5b00077.

[11] Melissa K. Takahashi, James Chappell, Clarmyra A. Hayes, Zachary Z. Sun, Jongmin Kim, Vipul Singhal, Kevin J. Spring, Shaima Al-Khabouri, Christopher P. Fall, Vincent Noireaux, Richard M. Murray, and Julius B. Lucks. Rapidly Characterizing the Fast Dynamics of RNA Genetic Circuitry with Cell-Free Transcription–Translation (TX-TL) Systems. *ACS Synthetic Biology*, March 2014. doi: 10.1021/sb400206c. URL http://dx.doi.org/10.1021/sb400206c.

[12] Jonathan Garamella, Ryan Marshall, Mark Rustad, and Vincent Noireaux. The All E. coli TX-TL Toolbox 2.0: A Platform for Cell-Free Synthetic Biology. *ACS Synthetic Biology*, 5(4):344–355, April 2016. doi: 10.1021/acssynbio.5b00296. URL http://dx.doi.org/10.1021/acssynbio.5b00296.

[13] Eric Walter and Yves Lecourtier. Global approaches to identifiability testing for linear and nonlinear state space models. *Mathematics and Computers in Simulation*, 24(6):472–482, December 1982. ISSN 0378-4754. doi: 10.1016/0378-4754(82)90645-0. URL http://www.sciencedirect.com/science/article/pii/0378475482906450.

[14] Vipul Singhal, Zoltan A Tuza, Zachary Z Sun, and Richard M Murray. A MATLAB toolbox for modeling genetic circuits in cell-free systems†. *Synthetic Biology*, 6(1):ysab007, October 2021. ISSN 2397-7000. doi: 10.1093/synbio/ysab007. URL https://doi.org/10.1093/synbio/ysab007.

[15] Yutaka Hori and Richard M. Murray. A state-space realization approach to set identification of biochemical kinetic parameters. In *Control Conference (ECC), 2015 European*, pages 2280–2285. IEEE, 2015. URL http://ieeexplore.ieee.org/abstract/document/7330878/.

[16] E. August. Parameter Identifiability and Optimal Experimental Design. In *International Conference on Computational Science and Engineering, 2009. CSE '09*, volume 1, pages 277–284, August 2009. doi: 10.1109/CSE.2009.39.

[17] Alexander Holiday, Mahdi Kooshkbaghi, Juan M. Bello-Rivas, C. William Gear, Antonios Zagaris, and Ioannis G. Kevrekidis. Manifold learning for parameter reduction. *Journal of Computational Physics*, 392:419–431, September 2019. ISSN 0021-9991. doi: 10.1016/j.jcp.2019.04.015. URL https://www.sciencedirect.com/science/article/pii/S0021999119302487.

[18] Hector J. Sussmann. Existence and uniqueness of minimal realizations of nonlinear systems. *Mathematical systems theory*, 10(1):263–284, December 1976. ISSN 1433-0490. doi: 10.1007/BF01683278. URL https://doi.org/10.1007/BF01683278.

[19] Velimir Jurdjevic. Abstract Control Systems: Controllability and Observability. *SIAM Journal on Control*, 8(3):424–439, August 1970. ISSN 0036-1402. doi: 10.1137/0308030. URL https://epubs.siam.org/doi/abs/10.1137/0308030. Publisher: Society for Industrial and Applied Mathematics.

[20] Héctor J Sussmann and Velimir Jurdjevic. Controllability of nonlinear systems. *Journal of Differential Equations*, 12(1):95–116, July 1972. ISSN 0022-0396. doi: 10.1016/0022-0396(72)90007-1. URL https://www.sciencedirect.com/science/article/pii/0022039672900071.

[21] R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, 22(5):728–740, October 1977. ISSN 1558-2523. doi: 10.1109/TAC.1977.1101601. Conference Name: IEEE Transactions on Automatic Control.

[22] A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, August 2009. ISSN 1367-4803. doi: 10.1093/bioinformatics/btp358. URL https://academic.oup.com/bioinformatics/article/25/15/1923/213246/Structural-and-practical-identifiability-analysis.

[23] Zachary Z. Sun, Clarmyra A. Hayes, Jonghyeon Shin, Filippo Caschera, Richard M. Murray, and Vincent Noireaux. Protocols for Implementing an Escherichia coli Based TX-TL Cell-Free Expression System for Synthetic Biology. *Journal of Visualized Experiments : JoVE*, (79), September 2013. ISSN 1940-087X. doi: 10.3791/50762. URL http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3960857/.

[24] Zachary Z. Sun, Enoch Yeung, Clarmyra A. Hayes, Vincent Noireaux, and Richard M. Murray. Linear DNA for Rapid Prototyping of Synthetic Biological Circuits in an Escherichia coli Based TX-TL Cell-Free System. *ACS Synthetic Biology*, 3(6):387–397, June 2014. doi: 10.1021/sb400131a. URL http://pubs.acs.org/doi/abs/10.1021/sb400131a.