Chapter 2

A Model-Based Calibration Methodology for Cell-Free Extract Variability Reduction

2.1 Introduction

Cell-free extracts have been proposed as a potential tool for the rapid prototyping of genetic circuits in synthetic biology [47]. One of the challenges in the development of this technology is that there is significant variability across different batches of extracts, which limits our ability to reliably generalize the results of any one extract. The study performed by Takahashi et al. [65] showed large variation in the constitutive expression of a fluorescent protein between batches, and Hu et al. [33] went one step further, showing that the variability in expression could be mapped to variability in the parameter estimates. Interestingly, Garamella et al. [22] showed minimal variability in constitutive gene expression between four extract batches. However, these batches were produced by the same expert personnel under strictly controlled conditions and supervision (personal communication), and such reproducibility has not been demonstrated in other labs. Furthermore, Garamella et al. did not demonstrate the lack of variability in the behavior of more complex circuits.

Our main goal in this chapter is to describe a method for computationally correcting for the variability between extracts. We frame the reduction in batch-to-batch variability in terms of what we call the *data correction problem*. This involves finding a method for

transforming the behavior of a circuit from a *candidate* extract into what it would look like had it been collected in a *reference* extract. The idea is that whenever data are collected in a batch of extract, they should be transformed into their reference extract version, making them directly comparable with other similarly transformed data.

The data correction problem may be solved by a model-based methodology for calibrating extract batches and subsequently using these calibrations to correct measured genetic circuit behavior. We call this procedure the *calibration-correction* method, after an analogous procedure developed for correcting wind tunnel data in the 1940s [63]. The assumption underlying the method is that there are certain features of extracts that vary from batch to batch, and the variation of these features can be captured as the variation of certain *extract specific parameters* (ESPs). Furthermore, we assume the parameters associated with the circuit do not change when implemented in different extract batches. In general, this assumption should hold for sufficiently fine grained circuit models, becoming more approximate when coarser models are used.

The calibration-correction method involves first performing a set of calibration experiments on both the reference and candidate extracts, and using models corresponding to these experiments to estimate the ESPs associated with each extract. Subsequently, the behavior of a circuit of interest is measured in the candidate extract, and its *circuit specific parameters* (CSPs) are estimated from this data and a corresponding model. This estimation step is performed with the ESPs for the candidate extract fixed at the values obtained at the calibration stage. Finally, the prediction for the circuit behavior in the reference extract is generated using the circuit model with these CSPs, along with the ESPs for the reference extract.

The choice of ESPs and CSPs is hypothesis driven, and must be verified experimentally. In this work we assume that extract specific parameters generally correspond to parameters associated with cellular machinery like RNA polymerases and ribosomes, while the CSPs correspond to parameters like transcription factor dimerization constants, which are associated primarily with circuit parts.

The implementation of this method is complicated by the fact that the parameters

of the models we use to describe biochemical systems are rarely completely identifiable. Roughly stated, parameter non-identifiability refers to the situation when the parameter estimation inverse problem is underdetermined, leading to non-unique solutions. It occurs when the data are not sufficiently informative for the level of detail present in the model. Set based parameter estimation methods like that in [32] or MCMC allow for the computation of equivalence classes [66] of parameter values that fit the model behavior to the data. The fact that the parameter sets obtained are equivalence classes with respect to a model-data set pair simply means that one may sample an *arbitrary* point from the identified parameter set, and it will be a point at which the model fits the data. This notion leads to the question of whether these sets of parameters can be treated as equivalence classes with respect to any method that depends on solving the inverse problem, which will be a major theme in this chapter.

The main conceptual contribution of this chapter will be to show that with respect to the calibration-correction problem, and under some consistency conditions on the parameter sets, these sets can indeed be treated as equivalence classes. The statement of the conditions on the parameter sets will lead to a prescription of how to design the calibration experiments, and in Section 2.6, even lead to a refinement of the calibration-correction method itself.

The framework presented in this work is not limited to correcting the behavior of genetic circuits across cell extracts, and may be applied to the correction of behavior between different cell strains, between the *in vitro* and *in vivo* environments, and even to applications in other engineering disciplines [63]. As such, even though we continue to refer to circuits and extracts, we note that replacing these with *process* and *environment* respectively allows this framework to be used elsewhere.

We start by showing that extracts prepared using the same protocol by different individuals display large variability in gene expression. We then define some notation in Section 2.3. In Section 2.4 we describe the data correction problem and the calibration-correction method in formal terms, and demonstrate the method using a simple example. In Section 2.5, we discuss a set of conditions that are required to hold for the method

to work in general, and discuss the limitations of the method in light of these conditions. Section 2.6 introduces a refinement of the method that improves its performance, and discuss its effect on our example system. In Section 2.7, we demonstrate how the refinement works on artificial data, and end with some concluding remarks in Section 2.8.

2.2 Extracts Display Significant Variability Across Batches

We start with a demonstration of the variability in extract behavior in three batches of extracts. The extracts used were created using the protocol in [60] using the BL21 Rosetta bacterial strains, with cell lysis performed using a French press instead of the bead-beating method described there. Example data in these extracts are shown in Figure 2.1, which shows the results of expressing six constitutive transcriptional units in the three extracts, expressed with linear DNA and GamS protein for protection from nucleases. The buffer used for the experiments in Figure 2.1 was the same. The experiments were performed with five technical repeats, and the mean and standard deviation (shown as solid lines and shaded regions in corresponding colors in the figure) were computed.

We note that the extracts show different levels of expression across different promoters. In particular, eJP shows the highest expression across all the constructs, followed by eVS, and finally eSG.

2.3 Notation and Preliminary Ideas

2.3.1 Experiments, Systems, Models and Parameters

We consider systems $\mathcal{S} = (\mathcal{E}, \mathcal{C})$ described as a combination of an extract $\mathcal{E}$ and a circuit $\mathcal{C}$, 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 $\overline{y}$. The bar symbol ($\bar{}$) over y is used to denote the fact that the experimental data are assumed to reflect the true behavior of the system, as opposed to the model output trajectories $\hat{y}$ generated at a particular parameter point and set of initial conditions. The definitions of the data correction problem and the

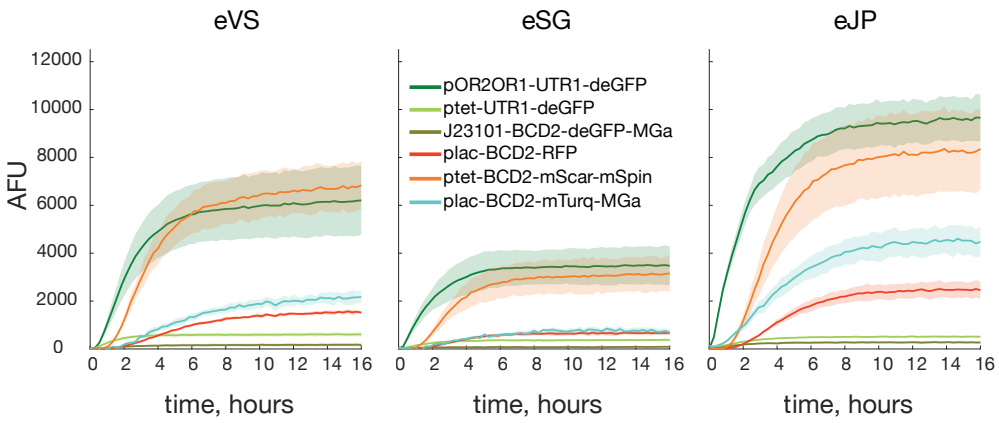

Figure 2.1: There is significant batch-to-batch variability in extracts. We expressed six constitutive reporter constructs (n = 5, technical repeats, shaded region = standard deviation) in three extract batches prepared by different scientists. Each of the constructs was expressed on linear DNA.

calibration-correction method, along with the conditions for the method to work will be given in the context of a model universe, where artificial data will be produced by models with known parameters. Here, the true data variable will also be denoted by $\overline{y}$, and will be distinguished from sample output trajectories $\hat{y}$ and the generic symbol y. For simplicity, we only allow for inputs in the form of initial conditions to the systems, though inputs at other times may be included without significant change to the framework or the mathematical results derived in this chapter.

The parameter vector θ of a model M associated with a given experiment will be partitioned into *extract specific parameter* (ESP) coordinates $e \in \mathbb{R}^{q_E}$, and *circuit specific parameter* (CSP) coordinates $c \in \mathbb{R}^{q_C}$. We do not restrict these parameters to be in the positive orthant, since any positive parameters may be log transformed to exist in the entire space. In subsequent definitions and proofs, we will declutter notation by dropping the explicit specification of the spaces these parameters live in, but these will always be assumed to be as defined here.

The partition of $\theta = (e, c)$ into ESPs and CSPs may be made using the following guidelines: ESPs are parameters associated primarily with species that are present in the system regardless of the the circuit implemented. Examples of ESPs are the concentration of transcriptional and translational machinery, elongation rates for transcription and translation

and the concentration of RNA degradation machinery. CSPs are parameters associated with species that may no longer exist in the system when the circuit is changed. Examples include promoter-transcription factor binding parameters or transcription factor dimerization parameters.

We define an initialized parametrized model with the equations

$$\begin{aligned} \dot{x} &= f(x,\theta), \\ y(\theta,x_0) &= h(x,\theta), \qquad x(0) = x_0(\theta). \end{aligned} \tag{2.1}$$

Here the state vector and its initialization are $x, x_0 \in \mathbb{R}^n_+$ respectively, the solutions are assumed to exist for all $t \geq 0$, the parameter vector symbol is $\theta = (e,c) \in \Omega$, where Ω is the set of all parameter values of interest. The hat symbol (ˆ) over a parameter denoting an estimated value of the parameter, as in $\hat{\theta}$ or $\hat{e}$. We shall reserve the tilde (˜) symbol for miscellaneous purposes, such as picking an arbitrary point in a set while proving an assertion. The output is denoted $y(t,\theta,x_0) \in \mathbb{R}^r$. For simplicity, we do not explicitly model inputs to the system, finite intervals of existence of solutions, or restrictions of the state and parameter spaces to sets smaller than the non-negative orthant. The overall mathematical framework and arguments we develop do not depend on these simplifications, and the general case can be included if needed. The functions f and h are assumed to be analytic vector fields with respect to x in some neighborhood of any attainable x [69]. Lastly, time dependence of the vector fields can be modeled by including t in the state variables. We will use the shorthand $y(\theta,x_0) = M(\theta,x_0)$ to refer to a model in Equation (2.1). We will almost always simplify this notation by dropping the explicit dependence on x_0, which will be assumed to be implicit and appropriately defined in every model. Furthermore, we will often replace θ with (e,c), as in stating $y(e,c) = M(e,c)$ or $M(e,c)$, instead of $y(\theta) = M(\theta)$ where appropriate.

2.3.2 Model Universe

Our analytical results will be stated and proved in a virtual *model universe*, where artificial data $\overline{y}$ is generated using models (denoted by $\overline{M}$) with known nominal parameter values

($\overline{\theta}$). The primary reason for this is that our theoretical results can only be proven when one has access to the true values of the parameters that generated the data.

Furthermore, we will limit ourselves to the case where the models used to estimate parameters from the data are the very models used to generate the data in the first place. In particular, both models will have the same dynamical equations f specifying them. In this sense, the models we use to work with the data are correct, and this assumption will be denoted by $M = \overline{M}$. 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, i.e., without also having to be concerned with whether our models are good models of the system that generated the data. Issues associated with model correctness or the use of increasingly approximate models (that often arise due to model order reduction) are left as future extensions of this work. Furthermore, it is worth explicitly stating that even though the nominal parameter vector used to generate the output trajectory is a single point in the parameter space, 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 2.5, 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.

More precisely, when we refer to a model universe, we identify an experiment $\mathcal{H} = (\mathcal{S}, x_0, \overline{y})$ with a model with known nominal parameters, $\overline{y} = \overline{M}(\overline{\theta}, x_0)$. Here, $\overline{y}$ denotes the measurements from these virtual experiments in our model universe.

2.3.3 Parameter Non-Identifiability

In this subsection, we follow Walter and Lecourtier [69] in defining the notion of parameter non-identifiability.

Definition 1 (Output-Indistinguishable)**.** 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

$$\begin{aligned} &\theta_A,\ \theta_B \in \Omega, \\ &y(\theta_A, x_0) = y(\theta_B, x_0) \quad \forall t \geq 0,\ \forall x_0 \in \mathbb{R}^n_+. \end{aligned} \tag{2.2}$$

Definition 2 (Structural Global Identifiability (parameter))**.** The i^{th} coordinate of θ_A, denoted $\theta_{A,i}$, is *structurally globally identifiable* (SGI) if for almost any $\theta_A \in \Omega$, Equation (2.2) has a unique solution for $\theta_{B,i}$.

This means that the i^{th} coordinate of the parameter vector being SGI 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 SGI coordinate, output indistinguishable trajectories almost always lead to a unique estimate of the coordinate.

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

The key point to note is that 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 to achieve a modeling objective such as ours. To this end, we shall consider models with non-SGI parameters, and thus allow e and c to exist in sets of output-indistinguishable parameters, denoted by E and C respectively.

2.3.4 Reference and Candidate Extracts, Calibration and Test Circuits

We define two extracts, the *reference extract* ($\mathcal{E}_1$), and a *candidate extract* ($\mathcal{E}_2$). Let $\mathcal{H}_{i,\text{cal}}$ be an experiment performed with a *calibration circuit*, $\mathcal{C}_{\text{cal}}$, on an extract $\mathcal{E}_i$ to determine the extract specific parameters, and let $\mathcal{H}_{i,\text{test}}$ be an experiment carried out with a *test circuit*, $\mathcal{C}_{\text{test}}$. The goal of the data transformation is to carry out the test circuit experiment in the candidate extract, $\mathcal{H}_{2,\text{test}} = (\mathcal{S}_{2,\text{test}}, x_{0,\text{test}}, \overline{y}_{2,\text{test}})$, and transform output measurements

towards what they would have looked like had they been collected in the reference extract, i.e., transform $\overline{y}_{2,\text{test}}$ into $\hat{y}_{1,\text{test}} \approx \overline{y}_{1,\text{test}}$ where $\overline{y}_{1,\text{test}}$ is the output of $\mathcal{H}_{1,\text{test}}$. Our overall strategy will be to use the estimates of the ESPs obtained at the calibration step in the test circuit models at the correction step. This will require that the models used to describe the calibration and test circuits be similar in the sense that they model the core processes at the same level of abstraction. In the example in Section 2.4.3, protein production is modeled using a single enzymatic reaction.

2.4 A Calibration-Correction Methodology Can be Used to Reduce Extract Variability

In this section, we define the calibration-correction methodology, and demonstrate it using an example. As mentioned in the previous section, we are interested in framing the methodology in a manner that allows for the non-identifiability of model parameters. In Sections 2.4.1 and 2.4.2, we give formal definitions of the data correction problem, the parameter identification operation and the calibration-correction method, and in Section 2.4.3, we demonstrate it on a simple example of correcting the behavior of a tetR mediated repression system.

2.4.1 Framing Extract Variability Reduction as the Data Correction Problem

We begin by framing the variability reduction problem in terms of the data correction problem, defined below and shown schematically in Figure 2.2.

Definition 4 (The Data Correction Problem)**.** Let $\mathcal{H}_{i,\text{test}} = ((\mathcal{E}_i, \mathcal{C}_{\text{test}}), x_{0,\text{test}}, \overline{y}_{i,\text{test}})$, $i = 1, 2$, 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 $\bar{e}_1$ and $\bar{e}_2$ be the ESPs for $\mathcal{E}_1$ and $\mathcal{E}_2$ respectively. Let $\bar{c}_{\text{cal}}$ and $\bar{c}_{\text{test}}$ be the CSPs for the calibration and test experiments respectively. Then the output data we discuss in the model universe is

$$\overline{y}_{i,\text{cal}} \triangleq \overline{M}_{\text{cal}}(\bar{e}_i, \bar{c}_{\text{cal}}), \tag{2.3}$$

$$\overline{y}_{i,\text{test}} \triangleq \overline{M}_{\text{test}}(\bar{e}_i, \bar{c}_{\text{test}}), \tag{2.4}$$

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, $d(\overline{y}_{1,\text{test}}, \hat{y}_{1,\text{test}}) < d(\overline{y}_{1,\text{test}}, \overline{y}_{2,\text{test}})$. ◇

2.4.2 The Calibration-Correction Method as the Solution to the Data Correction Problem

We begin by describing the parameter identification as a set valued operation on a data-model pair, and subsequently use this as a basis for defining a sequence of steps that together constitute the calibration-correction method.

Definition 5 (Parameter Identification)**.** Let the set Γ 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 $\text{ID}_\theta : \Gamma \to \mathcal{P}(\Omega)$, with $\text{ID}_\theta(y, M(\theta)) = \{\hat{\theta} \in \Omega \mid y = M(\hat{\theta})\}$

In the definition above we have included θ as a subscript to the parameter identification operator to make explicit exactly which parameter coordinates within the model M are being identified. This helps with stylistic uniformity in the usage of this operator, because we also define a *conditional* version for it, where certain parameter coordinates

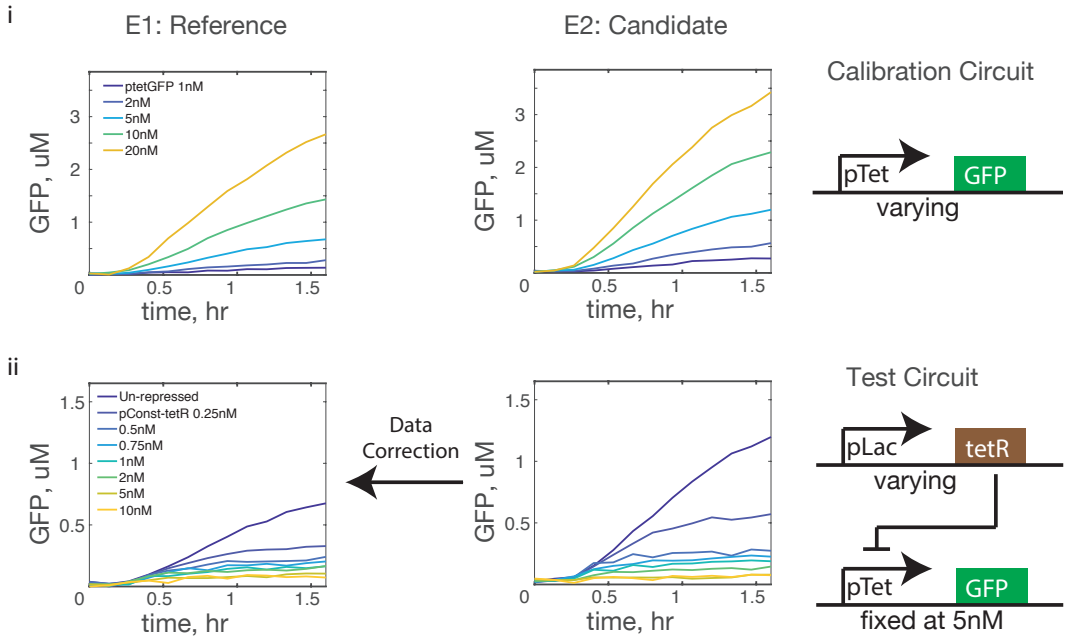

Figure 2.2: The Data Correction Problem. The data correction problem of Definition 4 involves the transformation of the behavior of a genetic circuit, which we refer to as the *test* circuit (ii), from a *candidate* extract to a *reference* extract. We have the freedom to design and implement a set of *calibration* experiments ($\mathcal{H}_{i,\text{cal}}$, $i = 1, 2$) on the two extracts (i), and collect the resulting data ($\overline{y}_{1,\text{cal}}$ and $\overline{y}_{2,\text{cal}}$). In this figure, the test circuit is the repression of the pTet promoter by constitutively expressed tetR transcription factor. For the calibration experiments, we will demonstrate that simply using constitutive GFP expression is sufficient to transform the data when the parameter non-identifiability is addressed using the tools developed in this work.

are held at fixed values, and sets of values of the remaining coordinates are estimated. This is discussed in Remark 3 next.

Remark 3. We define two minor modifications to the use of the ID_θ operator. First, we allow for the identification of a subset of parameter coordinates, such as the circuit specific parameters c, with values for the remaining parameter coordinates fixed at a specified value. We use the notation $\mathrm{ID}_{c|e=\hat{e}}(y, M(e,c))$ or $\mathrm{ID}_{c|e=\tilde{e}}(y, M(e,c))$ to describe this version of the operator. Here, the hat or tilde is used to denote the fact that the ESP coordinates are set to a specific value ($\hat{e}$ or $\tilde{e}$), while the set of values for the remaining parameter coordinates (the CSP coordinates in this example) is free to be estimated by the operator. We also note that in this case, the domain and codomain of this operator are slightly different from those shown in the definition above. Indeed, the domain for the $\mathrm{ID}_{c|e=\tilde{e}}(y, M(e,c))$ case is the set of all pairs $(y, M(\tilde{e},c))$ for which there exists a parameter $\hat{c}$ such that $y = M(\tilde{e},\hat{c})$. Similarly, the codomain can be $\mathcal{P}(\mathbb{R}^{q_c})$ or $\mathcal{P}(\mathrm{proj}_c\,\Omega)$, where proj_c denotes the projection operator from the full coordinate space to the CSP coordinates, c.

In the rest of this chapter, we will simplify notation by shortening $\mathrm{ID}_{c|e=\tilde{e}}(y, M(e,c))$ to $\mathrm{ID}_c(y, M(\tilde{e},c))$. This should not cause any ambiguity, since both the c subscript and the tilde over the ESP coordinates e are being used to denote the fact that we are estimating the CSP coordinates c, while holding the ESP coordinates at $\tilde{e}$. In both cases, we call this the conditional ID operator.

A second method of identifying values for some subset of parameter coordinates (say c once again) is to identify values over all the parameter coordinates, and then to project the resulting set down to the coordinates of interest. An example of the notation we will use to describe this operation is $\mathrm{proj}_c\,\mathrm{ID}(y, M(\theta))$, where $\theta = (e,c)$ and the ID operator works on the full parameter vector θ, as defined in Definition 5. ◇

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 identified or used, and single trajectories generated, we also give descriptions of the sets of all such points and trajectories. The definitions of these sets allow for the investigation of the idea of whether the non-identifiable parameter

sets can be treated as equivalence classes with respect to this method. In particular, in Section 2.5, 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 2.3 shows a schematic description of this procedure.

Definition 6 (The Calibration-Correction Method)**.** Consider the data correction problem in the context of the model universe. We define the *calibration-correction method* as a sequence of steps 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 prediction of the behavior of the test circuit in the reference extract, denoted by $\hat{y}_{1,\text{test}}$. The steps are:

1. *Calibration Step.* Find extract specific parameters that fit the calibration model to corresponding data for each of the extracts, while sharing a common estimate of the circuit specific parameter vector. I.e., find $\hat{e}_{1,\text{cal}}$ and $\hat{e}_{2,\text{cal}}$ such that the tuple $(\hat{e}_{1,\text{cal}}, \hat{e}_{2,\text{cal}}, \hat{c}_{\text{cal}})$ satisfies $\overline{y}_{1,\text{cal}} = M_{\text{cal}}(\hat{e}_{1,\text{cal}}, \hat{c}_{\text{cal}})$ and $\overline{y}_{2,\text{cal}} = M_{\text{cal}}(\hat{e}_{2,\text{cal}}, \hat{c}_{\text{cal}})$ for some $\hat{c}_{\text{cal}}$. Note that the set of all such ESP points is constructed as follows: first, the set of all valid $(\hat{e}_{1,\text{cal}}, \hat{e}_{2,\text{cal}}, \hat{c}_{\text{cal}})$ tuples is defined as

$$\tilde{\Theta}_{\text{cal}} \triangleq \Big\{ (e_1, e_2, c) \,\Big|\, \overline{y}_{i,\text{cal}} = M_{\text{cal}}(e_i, c),\, i = 1, 2 \Big\},$$

   and then, the ESP sets are defined as

$$E_{i,\text{cal}} \triangleq \text{proj}_{e_i} \tilde{\Theta}_{\text{cal}}, \qquad i = 1, 2. \tag{2.5}$$

2. *Correction Step One.* Identify circuit specific parameters of the test circuit in the candidate extract while holding the extract specific parameters at the value estimated at the previous step. I.e., find $\hat{c}_{2,\text{test}}$ such that $\overline{y}_{2,\text{test}} = M_{\text{test}}(\hat{e}_{2,\text{cal}}, \hat{c}_{2,\text{test}})$. Note that the set of all such points is given by

$$C'_{2,\text{test}} \triangleq \bigcup_{\hat{e} \in E_{2,\text{cal}}} \text{ID}_{c|e=\hat{e}} \Big(\overline{y}_{2,\text{test}}, M_{\text{test}}\big(e, c\big) \Big), \tag{2.6}$$

   where we have used the full notation for the conditional ID operator.

3. *Correction Step Two.* Predict test circuit behavior in the reference extract using the circuit specific parameters estimated in the first correction step, and extract specific parameters estimated in the calibration step. I.e., generate the prediction $\hat{y}_{1,\text{test}} = M_{\text{test}}(\hat{e}_{1,\text{cal}}, \hat{c}_{2,\text{test}})$. Note that the set of all predictions that can be generated is given by

$$Y_1 \triangleq \bigcup_{\hat{e}\in E_{1,\text{cal}}} \bigcup_{\hat{c}\in C'_{2,\text{test}}} \hat{y}_1(\hat{e},\hat{c}), \tag{2.7}$$

where individual predictions of the reference trajectories are given by $\hat{y}_1(\hat{e},\hat{c}) = M_{\text{test}}(\hat{e},\hat{c})$.

Remark 4. If the ESP sets from the calibration step were to be estimated, the version of the calibration step defined above would be straightforward to implement computationally. This is because the estimation of $\tilde{\Theta}_{\text{cal}}$ can be done in a single step (see Chapter 3, Section 3.5 for concurrent parameter inference tools), and the sets $E_{i,\text{cal}}$, for $i = 1, 2$, are simple projections computed from the estimated set.

We also give an equivalent, but less computationally tractable 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 CSPs estimated in the two extracts. We start with estimating the joint ESP-CSP 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 CSPs where these agree, $C_{\text{cal}} \triangleq \text{proj}_c\,\Theta_{1,\text{cal}} \cap \text{proj}_c\,\Theta_{2,\text{cal}}$. Finally, the ESP sets are generated by restricting the $\Theta_{i,\text{cal}}$ by C_{cal},

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

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 e and c live in (i.e., q_E and q_C) are large enough that estimating $\tilde{\Theta}_{\text{cal}} \in \mathbb{R}^{2q_E+q_C}$ might be much more difficult compared to $\Theta_{i,\text{cal}} \in \mathbb{R}^{q_E+q_C}$. The tradeoff here is that intersections and restrictions of sets represented by point clouds can be computationally difficult. Finally, the lemma in Appendix 2.A establishes the equivalence of this definition to the one given in Definition 6 (Equation 2.5). ◇

Remark 5. Note that the set $C'_{2,\text{test}}$ is a subset of the larger set $C_{2,\text{test}} \triangleq \text{proj}_c \, \text{ID}_\theta(\overline{y}_{2,\text{test}}, M_{\text{test}})$. Indeed, $C'_{2,\text{test}}$ is obtained from $C_{2,\text{test}}$ by only keeping the points whose corresponding e coordinate values were in the calibration set $E_{2,\text{cal}}$. We use $C'_{2,\text{test}}$ because in the first correction step, we identify c only after fixing the value of e to an arbitrary point within $E_{2,\text{cal}}$. ◇

Remark 6. We can define two *failure conditions* for the calibration-correction method that will be useful in deriving the main theoretical results of this chapter. Both the conditions must be avoided for the calibration-correction method to solve the data correction problem.

The first condition (FC1) 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. For example, in the first correction step, if $\hat{e}_{2,\text{cal}}$ is such that there is no $\tilde{c}$ that satisfies $\overline{y}_{2,\text{test}} = M_{\text{test}}(\hat{e}_{2,\text{cal}}, \tilde{c})$, then the parameter estimation step fails at this point. In terms of Equation (2.6), this failure condition occurs if it occurs for *any* point e in $E_{2,\text{cal}}$.

The second failure condition (FC2) occurs if correction step two is able to produce a trajectory not equal to the true trajectory, i.e., $\hat{y}_{1,\text{test}} \neq \overline{y}_{1,\text{test}}$. In terms of the set Y_1 defined in Equation (2.7), this means that Y_1 contains at least one element that is not equal to $\overline{y}_{1,\text{test}}$. ◇

Before we state and prove the conditions that need to hold for this method to work, we illustrate its use with a simple example.

2.4.3 A Simple Example

To illustrate the calibration-correction method, we use tetR mediated repression as our test circuit experiment, constitutive GFP expression as our calibration circuit experiment, and model protein production directly from DNA using an enzymatic reaction. Figure 2.4 shows the data, and the results from the calibration and correction steps. The test circuit experiment involves fixing the tetR repressible ptet-UTR1-deGFP DNA at $5\,\text{nM}$, and varying the constitutive tetR DNA concentration from 0–$0.75\,\text{nM}$. The calibration experiment

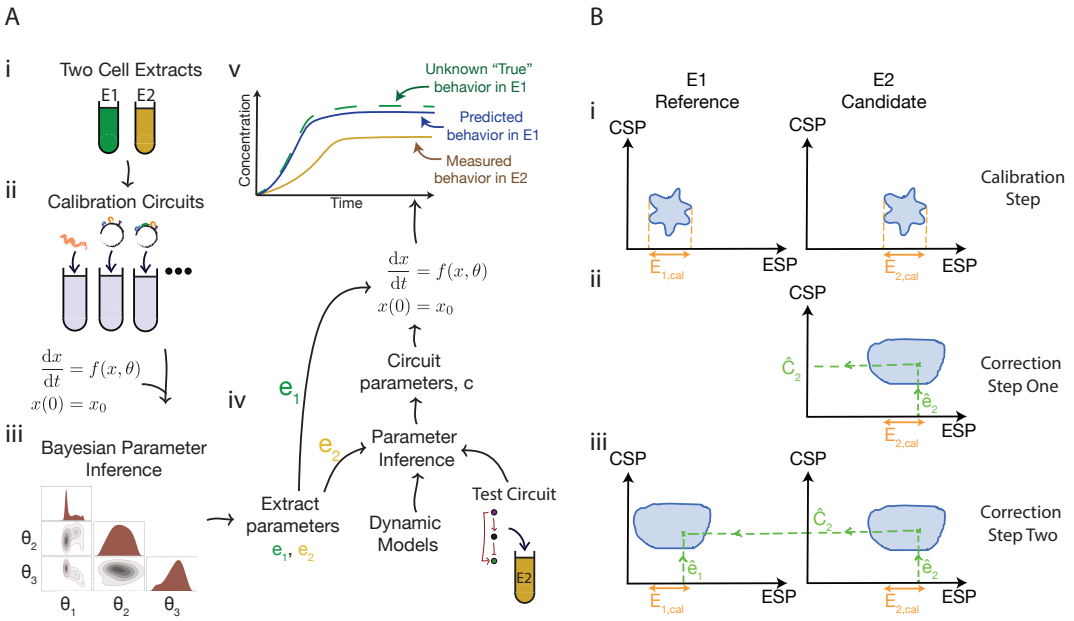

Figure 2.3: (A) Schematic describing the calibration-correction method of Definition 6. Calibration Step: Given two cell extracts (Ai), a reference extract $\mathcal{E}_1$ and a candidate extract $\mathcal{E}_2$, perform a set of calibration experiments (Aii) on each of the two extracts, and collect the corresponding data. Use parametrized models describing these experiments, along with parameter estimation tools (Aiii), to estimate the extract specific parameters (e_1 and e_2) as described in the calibration step of Definition 6. (Aiv) Correction Step One: Collect data for a test circuit in $\mathcal{E}_2$. The goal is to transform this into what it would look like had it been collected in $\mathcal{E}_1$. Use a model of the test circuit to estimate the CSPs for this circuit with the ESPs fixed to a value obtained for $\mathcal{E}_2$'s ESPs in the calibration step. The model used here must be at a similar level of detail as the models used for the calibration step. This allows for the ESPs estimated at that step to be used here. (Av) Correction Step Two: Finally, plug in the ESPs for $\mathcal{E}_1$ and the CSPs just estimated into the test circuit model to generate the desired transformed data (blue solid line) in the time-course schematic shown. (B) Interplay of parameter non-identifiability with the calibration-correction method. When the parameter estimation procedure returns sets of parameters that all fit the model to the data, we say that the parameters of the model are non-identifiable. (Continued below)

Figure 2.3: (Continued from above) The calibration-correction method in the presence of non-identifiability involves treating the sets of estimated parameters as equivalence classes, allowing for arbitrary points in the sets to be used for the purposes of the method. (Bi) Calibration step: The calibration step is still performed with the CSPs shared across the two extracts, and ESPs estimated individually, resulting in a set of points ((e_1, e_2, c)). In the schematic, the projections of this set onto the (e_1, c) and (e_2, c) coordinate axes are shown as the shaded regions. The ESPs that are obtained at the calibration step are now sets in the ESP coordinates, $E_{1,\text{cal}}$ and $E_{2,\text{cal}}$, corresponding to the projection of the sets in the full parameter space onto the e_1 and e_2 coordinate axes. (Bii) The first correction step involves picking an arbitrary point in the set $E_{2,\text{cal}}$ and estimating the set of CSPs that fit the test circuit model to the data at this point, and then treating this set as an equivalence class in turn and picking an arbitrary point from this set. The shaded region denotes the set of all parameters in the full coordinate space that fit the test circuit to the data. (Biii) Correction Step Two: An arbitrary point from the ESP set for the reference extract, $E_{1,\text{cal}}$, is picked, along with the arbitrarily picked point from the first correction step, and used to parameterize the test circuit model and generate the desired correction.

involves varying this reporter construct in isolation from $1\text{–}20\,\text{nM}$. The calibration circuit M_{cal} is modeled as

$$\mathrm{D_G} + \mathrm{Enz} \underset{k_{rG}}{\overset{k_{fG}}{\rightleftharpoons}} \mathrm{D_G{:}Enz} \xrightarrow{k_c} \mathrm{D_G} + \mathrm{Enz} + \mathrm{G}, \tag{2.8}$$

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, and G is the GFP protein. The test circuit is modeled using the equations M_{test},

$$\begin{aligned}
\mathrm{D_T} + \mathrm{Enz} &\underset{k_{rT}}{\overset{k_{fT}}{\rightleftharpoons}} \mathrm{D_T{:}Enz} \xrightarrow{k_c} \mathrm{D_T} + \mathrm{Enz} + \mathrm{T}, \\
\mathrm{D_G} + \mathrm{Enz} &\underset{k_{rG}}{\overset{k_{fG}}{\rightleftharpoons}} \mathrm{D_G{:}Enz} \xrightarrow{k_c} \mathrm{D_G} + \mathrm{Enz} + \mathrm{G}, \\
2\mathrm{T} &\underset{k_{r,dim}}{\overset{k_{f,dim}}{\rightleftharpoons}} \mathrm{T_2}, \\
\mathrm{D_G} + \mathrm{T_2} &\underset{k_{r,rep}}{\overset{k_{f,rep}}{\rightleftharpoons}} \mathrm{D_G{:}T_2},
\end{aligned} \tag{2.9}$$

where $\mathrm{D_T}$ is the DNA that codes for the tetR repressor protein (under the control of a constitutive promoter), T and $\mathrm{T_2}$ are the tetR protein monomer and dimer respectively. Note that the tetR dimer sequesters the GFP expressing DNA, $\mathrm{D_G}$, and in doing so, represses GFP.

Recall that the models used for the circuits at the calibration and correction stages have to be at the same levels of modeling to allow for ESPs estimated at the calibration stage to be used in the test model at the correction stage. In the example above, both the models produce protein using a single step enzymatic reaction, with the parameters θ partitioned into ESPs $e = (k_c, [\mathrm{Enz}]_0)$ and the CSPs $c = (k_{fT}, k_{rT}, k_{fG}, k_{rG}, k_{f,dim}, k_{r,dim}, k_{f,rep}, k_{r,rep})$. Here, $[\mathrm{Enz}]_0$ denotes the initial concentration of Enz. The main reason for picking this simple model for protein expression is that at this level of modeling, the number of parameters is small enough that the theoretical conditions we discuss can be visualized in three dimensions before being generalized to models with higher dimensional parameter spaces.

Continuing with our example, we next perform the calibration step of the method using an MCMC method (see Section 3.5) to estimate the posterior distribution of the parameters given the data, $\mathbb{P}\left(e_1, e_2, c_{cal} \mid \overline{y}_{1,cal}, \overline{y}_{2,cal}, M_{cal}\right)$. We note that the calibration circuit CSPs are estimated jointly over the two extract batches, i.e., the distribution above is that of the vector $(e_1, e_2, c_{\mathrm{cal}})$ such that the model $M_{\mathrm{cal}}(e_i, c_{\mathrm{cal}})$ fits the data $y_{i,\mathrm{cal}}$ simultaneously for both values of $i = 1, 2$. Figure 2.4 shows the model fits from this step, and the corner-plots showing pairwise projections of the joint parameter distributions of the (e_i, c_{cal}) coordinates for both $\mathcal{E}_1$ and $\mathcal{E}_2$.

To perform the first correction step we fixed the candidate extract ESP value at a single point drawn from $E_{2,\mathrm{cal}}$ and estimated $C_{2,\mathrm{test}}$. The model fits are shown in Figure 2.4. Fixing the ESP value to a point in $E_{1,\mathrm{cal}}$ and drawing 500 points from $C_{2,\mathrm{test}}$ to generate the corrected trajectories implements correction step two, and the results are shown in the third column in Figure 2.4 (iii).

To conclude this section, 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. 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. Formally, we write the metric as,

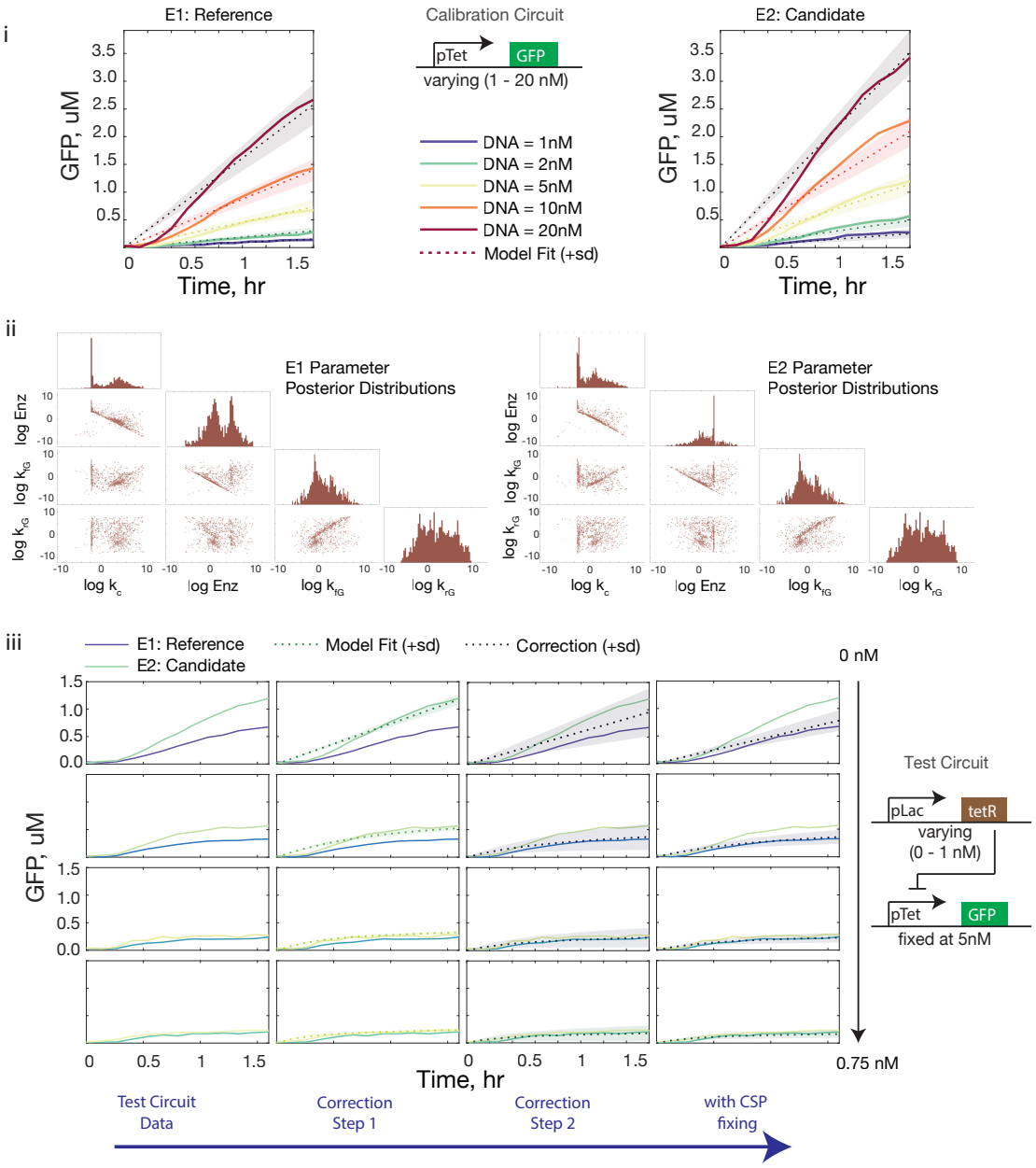

Figure 2.4: Demonstration of the calibration-correction method on the experimental data described in Figure 2.2. The calibration data are the constitutive expression of the pTet promoter at various DNA concentrations. The test data are the repression of a fixed concentration of the pTet promoter with varying concentrations of repressor DNA. (Continued below)

Figure 2.4: (continued from above) (i) Model fits to the calibration dataset using a Bayesian parameter inference approach. The joint parameter posterior distributions obtained using this approach are used as proxies for the parameter sets described in the main text. The model in Equation (2.8) is used with the calibration data to infer the joint posterior distributions of the ESPs and CSPs, denoted by the set $\tilde{\Theta}_{\text{cal}}$ in Definition 6. The solid lines depict the experimental data trajectories, and the dashed lines and shaded regions denote the means and standard deviations (resp.) of trajectories simulated using points drawn from the posterior distribution. (ii) Parameter posterior distributions from the calibration step. The ESPs are $e = (k_c, [\mathrm{Enz}]_0)$, where we let $[\mathrm{Enz}]_0$ denote the initial amount of the Enz species. The CSPs are $c = (k_{fG}, k_{rG})$, the binding-unbinding rate constants of the DNA to the Enz species. The parameter inference was performed with the CSPs shared across the extract, i.e., in a joint space with points (e_1, e_2, c). Here we show the posterior distributions of the parameter vector (e_i, c) for the two extracts $i = 1, 2$. The distributions are shown as corner plots of the pairwise projections on the off diagonal plots and the marginal distributions on the diagonal. (iii) The two correction steps on the tetR repression test circuit data. The reporter DNA is fixed at $5\,\mathrm{nM}$, and the repressor DNA varies from from 0–$0.75\,\mathrm{nM}$ down the rows (0, 0.25, 0.5 and $0.75\,\mathrm{nM}$). The solid lines in all the plots are experimental data, the dashed lines and shaded regions are the mean and standard deviations of simulated trajectories corresponding to parameters drawn from the respective parameter sets as described by the calibration-correction method in Definition 6. The first three columns, starting from the left, are: test circuit data in the two extracts, correction step one, where the model in Equation (2.9) is fit to the candidate extract data, and the second correction step. We see that the model fits the candidate extract data quite well in the first correction step, and correction step two is able to move the model prediction trajectories towards the reference extract data trajectories at all repressor DNA concentrations. The standard deviation of the predicted trajectories in the third column is much larger than that of the fitted trajectories in the second column. In Sections 2.6 and 2.7, we discuss ESP-CSP *covariation* as a possible reason for this type of increase in the standard deviation, and propose a modification to the calibration-correction method, called *CSP fixing*, that addresses this type of covariation. The fourth column shows the result of applying this modified version of the method to this data, and shows that the standard deviations tighten up considerably.

$$R_1 = \frac{\sum_{i=1}^{n_{IC}} \|\hat{y}_{1,\text{test}}(x_{0,i}) - \overline{y}_{1,\text{test}}(x_{0,i})\|_2}{\sum_{i=1}^{n_{IC}} \|\overline{y}_{2,\text{test}}(x_{0,i}) - \overline{y}_{1,\text{test}}(x_{0,i})\|_2}, \tag{2.10}$$

where the sum is taken over the n_{IC} experimental conditions (which, in this case, are the four tetR DNA concentrations). We have added an argument $\left(x_{0,i}\right)$ to the output trajectory variable y to reflect this fact explicitly. For our dataset, we compute the value of this metric to be $R_1 = 0.42$.

The second metric computes, for each of 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. Formally, it is defined as

$$R_2 = \frac{1}{n_{IC}} \sum_{i=1}^{n_{IC}} \frac{\|\hat{y}_{1,\text{test}}(x_{0,i}) - \overline{y}_{1,\text{test}}(x_{0,i})\|_2}{\|\overline{y}_{2,\text{test}}(x_{0,i}) - \overline{y}_{1,\text{test}}(x_{0,i})\|_2}, \tag{2.11}$$

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

2.5 Identifiability Conditions

In this section, we show that the SGI property is not necessary for the data correction problem to be solved by the calibration-correction method. This will be stated as a corollary of the main result of this section (Theorem 1), which gives conditions on the sets of non-identifiable parameters obtained during the calibration-correction method such that the method solves the data correction problem.

The key insight underlying the theory developed in this chapter is that since the correction only needs to be applied to the output trajectories, and not to the full state vector trajectories, we do not need the parameters to be fully identifiable. Roughly speaking, this is due to the fact that the non-identifiability occurs because the output trajectories are not informative enough to identify the parameters to a degree that allows the state tra-

jectories to be reconstructed. Indeed, the identified parameter sets only contain enough information to reconstruct the outputs. However, since we are only attempting to correct the outputs, and not the state trajectories, the method continues to work in the presence of the non-identifiability.

This idea of using parameters estimated using only the outputs to in turn correct only the output behavior, and not the entire state vector trajectories is closely related to the idea of the sets of output-indistinguishable parameters being equivalence classes with respect to the inputs and outputs of a model. While these sets may be equivalence classes with respect to individual estimations performed using model data 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. The main goal of this section will be to derive these conditions.

Theorem 1 (Parameter consistency)**.** *Consider the data correction problem (Definition 4) in the model universe, i.e., when the experimental data are generated by nominal parametrized initialized models, as described in Remark 1. Furthermore, consider the calibration-correction method of Definition 6, and the sets* $\tilde{\Theta}_{\text{cal}}$, $E_{1,\text{cal}}$, $E_{2,\text{cal}}$ *and* $C'_{2,\text{test}}$ *as defined there. 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,*

$$\tilde{\Theta}_{\text{cal}} \neq \emptyset, \tag{2.12}$$

$$E_{2,\text{cal}} \subseteq \text{proj}_e\, \Theta_{2,\text{test}}, \tag{2.13}$$

$$E_{1,\text{cal}} \times C'_{2,\text{test}} \subseteq \Theta_{1,\text{test}}, \tag{2.14}$$

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

Proof. We note that solving the data correction problem using the calibration-correction method simply involves avoiding the failure conditions FC1 and FC2 described in Remark 6. Avoiding FC1 wherever it may occur ensures that the method can be implemented in the first place, and avoiding FC2 means that the method returns the desired result. Thus, we must show that the conditions (2.12-2.14) are necessary and sufficient for avoiding FC1 and

FC2.

The necessity of condition (2.12) follows from the fact that if $\tilde{\Theta}_{\text{cal}} = \emptyset$, then there does not exist a vector (e_1, e_2, c) such that $\overline{y}_{i,\text{cal}} = M_{\text{cal}}(e_i, c)$ for $i = 1, 2$, leading to FC1 being met at the calibration step. While not needed for the proof, we note in passing that in the model universe, where $M_{\text{cal}}(\theta) = \overline{M}_{\text{cal}}(\overline{\theta})$ and $\overline{y}_{i,\text{cal}} = \overline{M}_{\text{cal}}(\overline{e}, \overline{c})$, condition (2.12) always holds.

Next, we prove the necessity of $E_{2,\text{cal}} \subseteq E_{2,\text{test}}$, where $E_{2,\text{test}} \triangleq \text{proj}_e \Theta_{2,\text{test}}$. Assume that there exists an $\tilde{e} \in E_{2,\text{cal}}$ such that $\tilde{e} \notin E_{2,\text{test}}$. Thus, there does not exist a $\tilde{c}$ such that $M_{\text{test}}((\tilde{e}, \tilde{c})) = \overline{y}_{2,\text{test}}$. Since the operator $\text{ID}_{c|e=\tilde{e}}$ is only defined on the set $\{(y, M) \mid \exists c : M((\tilde{e}, c)) = y\}$, we see that the map $\text{ID}_{c|e=\tilde{e}}(\overline{y}_{2,\text{test}}, M_{\text{test}}(e, c))$ is not well defined, leading to FC1 at the first correction step.

We prove the necessity of condition (2.14) as follows. Assume that there exists a $(\tilde{e}, \tilde{c}) \in E_{1,\text{cal}} \times C'_{2,\text{test}}$ such that $(\tilde{e}, \tilde{c}) \notin \Theta_{1,\text{test}}$. Since we use points $\hat{e} \in E_{1,\text{cal}}$ and $\hat{c} \in C'_{2,\text{test}}$ to generate the prediction $\hat{y}_{1,\text{test}}$ in the second correction step, it is possible that $\hat{e} = \tilde{e}$ and $\hat{c} = \tilde{c}$. Furthermore, since $\Theta_{1,\text{test}}$ is the set of all points (e, c) 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 (2.12-2.14) address both the points in the method where FC1 could be met, and the point in the method where FC2 could occur. Explicitly, condition (2.12) allows the calibration step to avoid FC1, condition (2.13) allows correction step one to avoid FC1, since it implies that for all $\tilde{e} \in E_{2,\text{cal}}$, there exists a $\tilde{c}$ such that $(\tilde{e}, \tilde{c}) \in \Theta_{2,\text{test}}$. Condition (2.14) enables correction step two to avoid FC2, since it implies that for all $\tilde{e} \in E_{1,\text{cal}}$ and for all $\tilde{c} \in C'_{2,\text{test}}$ we have that $\overline{y}_{1,\text{test}} = M_{\text{test}}(\tilde{e}, \tilde{c})$, implying that the set of all possible predicted trajectories only has the correct trajectory in it, $Y_1 = \{\overline{y}_{1,\text{test}}\}$. □

Remark 7. We can give some physical interpretations of the conditions (2.12-2.14). To do this, we first note that condition (2.14) implies (see Lemma 3 in Appendix 2.B)

$$E_{1,\text{cal}} \subseteq \text{proj}_e \Theta_{1,\text{test}}, \tag{2.15}$$

$$C'_{2,\text{test}} \subseteq C'_{1,\text{test}}, \tag{2.16}$$

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

$$C'_{1,\text{test}} \triangleq \bigcup_{\hat{e}\in E_{1,\text{cal}}} \text{ID}_{c|e=\hat{e}}\left(\overline{y}_{1,\text{test}}, M_{\text{test}}(e,c)\right).$$

Condition (2.12) and (2.15) may be interpreted to mean that the calibration experiments must be more informative about the ESPs than the test circuit experiments. This follows from the fact that the sets of output-indistinguishable ESPs obtained from the calibration step are subsets of the corresponding sets from the test circuits, $\text{proj}_e\,\Theta_{i,\text{test}}$.

Condition (2.16) says that the CSP sets for the test circuit, if estimated by first fixing the ESPs 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 (2.16).

Finally, condition (2.14) says that the ESP and CSP coordinates in the set $\Theta_{1,\text{test}}$ can only *covary* outside $E_{1,\text{cal}} \times C'_{2,\text{test}}$, i.e., all the points within this set must belong to $\Theta_{1,\text{test}}$. Covariation is defined in Section 2.6. ◇

Next, we state a few corollaries of the theorem.

Corollary 1 (SGI Sufficiency)**.** *SGI models are sufficient for the calibration-correction method to solve the data correction problem in the model universe.*

Proof. Recall from Remark 1 that in the model universe, the data are generated by nominal parameters, $\overline{e}_1$, $\overline{e}_2$, $\overline{c}_{\text{cal}}$, $\overline{c}_{\text{test}}$. We observe that since the models are SGI, these parameters uniquely fit the model to the data, and therefore the sets in conditions (2.12-2.14) only have single entries, leading to these conditions being trivially satisfied:

$$\tilde{\Theta}_{\text{cal}} = \{(\overline{e}_1, \overline{e}_2, \overline{c}_{\text{cal}})\} \neq \emptyset,$$
$$E_{2,\text{cal}} = \{\overline{e}_2\} \subseteq \text{proj}_e\{(\overline{e}_2, \overline{c}_{\text{test}})\} = \text{proj}_e\,\Theta_{2,\text{test}},$$
$$E_{1,\text{cal}} \times C'_{2,\text{test}} = \{\overline{e}_1\} \times \{\overline{c}_{\text{test}}\} \subseteq \{(\overline{e}_1, \overline{c}_{\text{test}})\} = \Theta_{1,\text{test}}.$$

□

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

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

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

Remark 8. We note that the condition $C'_{2,\text{test}} \subseteq C'_{1,\text{test}}$ discussed in Remark 7 gets transformed into $C'_{2,\text{test}} = C'_{1,\text{test}}$. ◇

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 = Calib' Case)**.** *Consider the data correction problem for the case where the test data and models are the same as the calibration data and models, i.e.,* $\overline{y}_{i,\text{test}} = \overline{y}_{i,\text{cal}}$ *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*

$$C'_{2,\text{cal}} \triangleq \bigcup_{\tilde{e} \in E_{2,\text{cal}}} \text{ID}_c\left(\overline{y}_{2,\text{cal}}, M_{\text{cal}}\left(\tilde{e}, c\right)\right). \tag{2.17}$$

Then, the conditions

$$\tilde{\Theta}_{\text{cal}} \neq \emptyset, \tag{2.18}$$

$$E_{2,\text{cal}} \subseteq \text{proj}_e \, \Theta_{2,\text{cal}}, \tag{2.19}$$

$$E_{1,\text{cal}} \times C'_{2,\text{cal}} \subseteq \Theta_{1,\text{cal}}, \tag{2.20}$$

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

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

2.6 Covariation Between ESP and CSP Parameter Coordinates Introduces Error into the Method

In this section, we describe covariation (Figure 2.5), 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 by defining a device that will be useful for taking slices of parameter sets.

Definition 7 (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 8 (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 $\tilde{\theta}_{a1}, \tilde{\theta}_{a2} \in \text{proj}_{\theta_a} \Theta$ such that $\text{proj}_{\theta_b}\left(\Theta \cap \text{cut}_{\theta_b}(\tilde{\theta}_{a1})\right) \neq \text{proj}_{\theta_b}\left(\Theta \cap \text{cut}_{\theta_b}(\tilde{\theta}_{a2})\right)$, then Θ is said to have *parameter covariation* of its θ_b coordinates with respect to its θ_a coordinates.

Remark 9. We will often abbreviate parameter covariation to just covariation, and say that parameter coordinates can *covary*. ◇

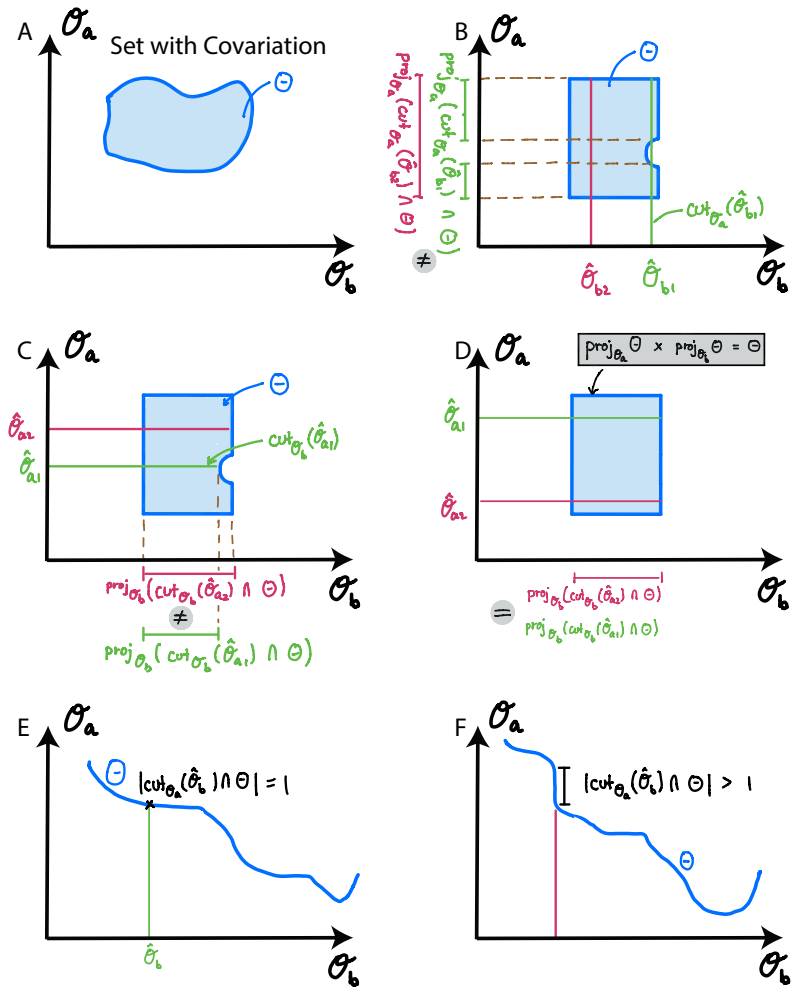

Figure 2.5: Schematic descriptions of parameter covariation and associated results. (A) The arbitrarily shaped set Θ shows parameter covariation. (B, C) Two ways of defining parameter covariation for a given set: Covariation of the θ_a coordinates with respect to the θ_b coordinates (B) and covariation of the θ_b coordinates with respect to the θ_a coordinates (C). The lines represent the cutting planes, and the intersection of these planes and the set Θ is projected onto the appropriate axes. Lemma 1 shows that covariation is equivalent to the Cartesian product condition in (D) not holding. This in turn can be used to show that the two ways of defining covariation (B and C) are equivalent, and therefore the definition of covariation is symmetric. (E, F) Thin covariation. (E) Thin covariation in the θ_a coordinates with respect to the θ_b coordinates. (F) The covariation in the θ_a coordinates is not thin with respect to the θ_b coordinates.

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. First, we prove the ($\Rightarrow$) direction. Covariation implies that for some $\theta_{a1}, \theta_{a2} \in \mathrm{proj}_{\theta_a} \Theta$ there exists a point $\tilde{\theta}_b \in \mathrm{proj}_{\theta_b} \Theta$ such that

$$\tilde{\theta}_b \in \left(\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a1})\right) \triangle \left(\mathrm{proj}_{\theta_b}\left(\Theta \cap \mathrm{cut}_{\theta_b}(\tilde{\theta}_{a2})\right), \tag{2.21}$$

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

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

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. □

Remark 10. 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 (e, c) 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 $E_{1,\mathrm{cal}} \times \mathrm{proj}_c \Theta_{2,\mathrm{cal}}$ has covariation with respect to the (e, c) partition.

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

tions

$$\tilde{\Theta}_{\text{cal}} \neq \emptyset, \tag{2.22}$$

$$C'_{2,\text{cal}} \subseteq \text{proj}_c\, \Theta_{1,\text{cal}}, \tag{2.23}$$

$$E_{i,\text{cal}} \subseteq \text{proj}_e\, \Theta_{i,\text{cal}}, \qquad i = 1,\, 2, \tag{2.24}$$

hold, but the set

$$\Theta'_{1,\text{cal}} \triangleq \Theta_{1,\text{cal}} \cap \left(E_{1,\text{cal}} \times \text{proj}_c\, \Theta_{2,\text{cal}}\right) \tag{2.25}$$

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

Proof. Condition (2.23), along with the fact that for the 'Test = Calib' case, $C'_{2,\text{cal}} = \text{proj}_c\, \Theta_{2,\text{cal}}$, implies that $\text{proj}_c\, \Theta'_{1,\text{cal}} = C'_{2,\text{cal}}$. Condition (2.24) implies $\text{proj}_e\, \Theta'_{1,\text{cal}} = E_{1,\text{cal}}$. Covariation implies that $\text{proj}_e\, \Theta'_{1,\text{cal}} \times \text{proj}_c\, \Theta'_{1,\text{cal}} \neq \Theta'_{1,\text{cal}}$. Thus, the proper subset relation $\Theta'_{1,\text{cal}} \subsetneq E_{1,\text{cal}} \times C'_{2,\text{cal}}$ holds, and therefore there exists $(\tilde{e}, \tilde{c}) \in E_{1,\text{cal}} \times C'_{2,\text{cal}}$ such that $(\tilde{e}, \tilde{c}) \notin \Theta'_{1,\text{cal}} \subseteq \Theta_{1,\text{cal}}$. This implies that $E_{1,\text{cal}} \times C'_{2,\text{cal}} \nsubseteq \Theta_{1,\text{cal}}$, which violates condition (2.20). □

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 = Calib' case when the CSP coordinates covary in this way with respect to the ESP coordinates. In Section 2.7.1, we will show that even the simplest models show non-identifiability with this type of covariation. We will also show that the variance blow up seen in the third column of Figure 2.4 decreases significantly when this modified version of the calibration-correction method is used.

Definition 9 (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 $\tilde{\theta}_b \in \text{proj}_{\theta_b}\, \Theta$, we have $\left|\text{cut}_{\theta_a}(\tilde{\theta}_b) \cap \Theta\right| = 1$, then we say that the covariation of the θ_a coordinates of Θ is thin with respect to the θ_b coordinates.

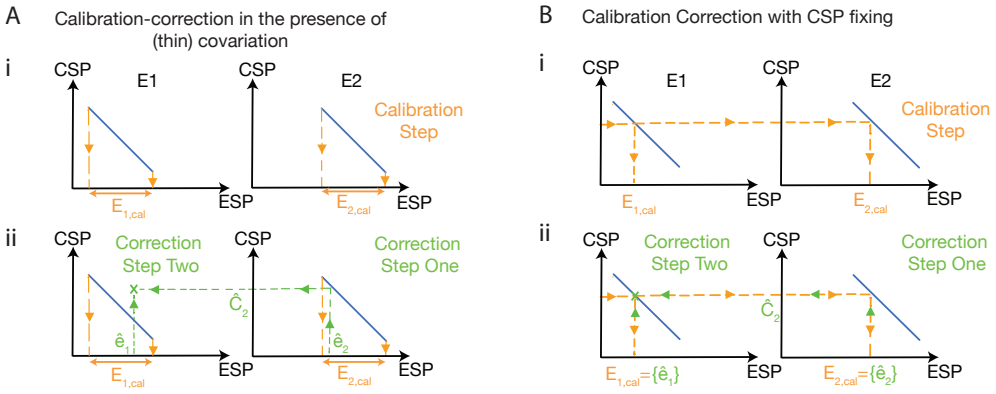

Figure 2.6: (A) A schematic description of how thin covariation between the ESP-CSP coordinates in the estimated joint parameter sets can cause calibration-correction to fail at correcting even the calibration data ('Test = Calib' special case described in Corollary 3). Columns correspond to extracts $\mathcal{E}_1$ and $\mathcal{E}_2$, rows to the calibration and correction steps, as labeled. The blue lines in all the plots are the joint ESP-CSP sets of all the parameter values that fit the calibration model to data. Covariation here is depicted by the fact that the blue line is not vertical, horizontal or a rectangle, i.e., as the CSP value changes, so does the ESP value, and so the ESP and CSPs cannot be picked independently from the respective projections onto the ESP and CSP coordinate axes. 'Thinness' of this covariation (of the CSP coordinates with respect to the ESP coordinates) corresponds to the fact that for each fixed ESP value in the set of possible values it can take, there is one and only one corresponding CSP value. (i) Under this setup, the calibration step leads to the ESP sets shown as projections of the blue lines. (ii) The first correction step fixes the ESP value to a point $\hat{e}_2 \in E_{2,\text{cal}}$, and estimates the only possible CSP value $\hat{c}_2$. The second correction step picks an arbitrary point $e_1 \in E_{1,\text{cal}}$, and uses the CSP value $\hat{c}_2$ to give the parameter point that will be used to generate the final predicted trajectory. It is clear that in general, due to covariation, this point will not lie on the blue line, which is the set of all points that will give the correct prediction. Indeed, this leads to the second failure condition (FC2) described in Remark 6. (B) How the CSP fixing modification (Definition 10) to the calibration step helps solve this issue. Consider the same setup as in (A), with the following exception: The ESP sets estimated at the calibration step are now generated by first intersecting the parameter sets (blue lines) with a line parallel to the ESP axis ('cutting plane' parallel to the ESP subspace in higher dimensions) centered at an arbitrary CSP value that can be attained (i.e., a value in the set $\mathbf{proj}_c\,\tilde{\Theta}_{\text{cal}}$), and secondly projecting these intersections to the ESP coordinates for both extracts. This CSP fixing modification is formally stated in Definition 10. It is clear that with this modification, following a logical procedure similar to the one in (A), the second correction step uses a parameter point on the blue line, avoiding FC2.

Remark 11. 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 SGI for each fixed θ_b. ◇

Remark 11 says 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 10 (CSP Fixing)**.** 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 $\tilde{c} \in \mathrm{proj}_c \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_c \Theta_{2,\mathrm{cal}}$. Then, we define CSP fixing as a modification to the calibration step in which the sets $E_{i,\mathrm{cal}} \triangleq \mathrm{proj}_e\left(\mathrm{cut}_e(\tilde{c}) \cap \Theta_{i,\mathrm{cal}}\right)$ 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 $\theta = (e, c)$. Assume that the $\Theta_{i,\mathrm{cal}}$ have thin covariation in their c coordinates with respect to their e coordinates. Then, the calibration-correction method with CSP fixing is able to solve the data correction problem for the 'Test = Calib' case of Corollary 3.*

Proof. Let $\tilde{c} \in \mathrm{proj}_c \Theta_{1,\mathrm{cal}} \cap \mathrm{proj}_c \Theta_{2,\mathrm{cal}}$ and $\tilde{e}_2 \in E_{2,\mathrm{cal}} \triangleq \mathrm{proj}_e\left(\mathrm{cut}_e(\tilde{c}) \cap \Theta_{2,\mathrm{cal}}\right)$. We note that the sets $\mathrm{proj}_c\left(\mathrm{cut}_c(\tilde{e}_2) \cap \Theta_{2,\mathrm{cal}}\right) = \mathrm{ID}_c(\overline{y}_{2,\mathrm{cal}}, M_{\mathrm{cal}}(\tilde{e}_2, c))$ are equal by definition. Now, pick an arbitrary point $\tilde{c}' \in \mathrm{proj}_c\left(\mathrm{cut}_c(\tilde{e}_2) \cap \Theta_{2,\mathrm{cal}}\right)$. It follows that $\tilde{c}' = \tilde{c}$ from the fact that $\tilde{c} \in \mathrm{proj}_c\left(\mathrm{cut}_c(\tilde{e}_2) \cap \Theta_{2,\mathrm{cal}}\right)$ and that the element in $\left|\mathrm{cut}_{\theta_a}(\tilde{\theta}_b) \cap \Theta\right| = 1$ is unique. Thus, the only possible CSP value that can be returned by the first correction step is $\tilde{c}$.

Next, we look at the second correction step. Pick an arbitrary $\tilde{e}_1 \in E_{1,\mathrm{cal}} \triangleq \mathrm{proj}_e\left(\mathrm{cut}_e(\tilde{c}) \cap \Theta_{1,\mathrm{cal}}\right)$. Since the point $(\tilde{e}_1, \tilde{c}) \in \Theta_{1,\mathrm{cal}}$, we have that $\overline{y}_{1,\mathrm{cal}} = \hat{y}_{1,\mathrm{cal}} \triangleq M(\tilde{e}_1, \tilde{c})$, and FC2 is avoided. □

2.7 Computational Investigation of Covariation and CSP fixing

In this section we investigate the effect of covariation on the calibration-correction method computationally, and show that CSP fixing helps reduce the error introduced by covariation. The general approach will be to generate artificial data using the models in Equations (2.8) and (2.9) 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

setting for the investigation, and are able to study the effects of non-identifiability without having to also consider issues of model correctness.

2.7.1 The 'Test = Calib' case of Corollary 3

We show that even the simplest models, such as that in Equation (2.8), show non-identifiability and (thin) covariation in this non-identifiability, and that the calibration-correction method of Definition 6 fails in the 'Test = Calib' special case of the data correction problem (Corollary 3) precisely in the way we expect from the theoretical framework developed in Section 2.6. We also show that with the CSP fixing modification to the calibration step, this type of failure is avoided.

We begin by generating artificial calibration data for extracts $\mathcal{E}_1$ and $\mathcal{E}_2$ using the calibration model in Equation (2.8) with the parameters in Table 2.1. The true trajectories are shown as dotted lines in Figure 2.7 (ii-iii). We have added a small amount of Gaussian noise to these trajectories for visualization purposes; however the trajectories used as data in the calibration-correction method do not contain this added noise. The calibration step was performed with $k_{fG} = 5$ fixed at its true value, reducing the number of parameters in the model to three (the sole CSP k_{rG}, and the pair of ESPs $[\mathrm{Enz}]_0$ and k_c) allowing for the visualization of the full joint distribution of the parameter samples that result from performing the MCMC parameter inference. This visualization is the most direct method of seeing the existence of non-identifiability and of thin covariation in the set of parameters

Table 2.1: Parameters Used to Generate Artificial Data

Type	Parameter	Extract 1 Value	Extract 2 Value	Model(s)
ESP	$[[\mathrm{Enz}]_0]$	100	200	M_{cal}, M_{test}
ESP	k_c	0.012	0.024	M_{cal}, M_{test}
CSP	k_{fG}	5	5	M_{cal}, M_{test}
CSP	k_{rG}	300	300	M_{cal}, M_{test}
CSP	k_{fT}	5	5	M_{test}
CSP	k_{rT}	300	300	M_{test}
CSP	$k_{f,dim}$	20	20	M_{test}
CSP	$k_{r,dim}$	10	10	M_{test}
CSP	$k_{f,rep}$	20	20	M_{test}
CSP	$k_{r,rep}$	10	10	M_{test}

that result from the parameter estimation.

The fitting of the model to the data (Figure 2.7 (ii, iii)) in the calibration step results in an estimate of the joint distribution of the parameter vector $(e_1, e_2, c) \in \tilde{\Theta}_{\text{cal}}$. The three dimensional scatter plots of empty blue circles in Figure 2.7 (iv, v) show the results of this estimation marginalized to the coordinates $(e_2, c) = ([\text{Enz}]_{0,2}, k_{c2}, k_{rG})$ and (e_1, c) respectively for the two extracts. We also fit a surface to the scattering of these points (translucent green gridded surface plot), which helps visualize the fact that these points essentially lie on a two dimensional surface within the three dimensional space of parameters, and that this surface displays thin covariation in its CSP coordinates with respect to its ESP coordinates. The calibration concludes with the projection of the points onto the ESP axes for $\mathcal{E}_1$ and $\mathcal{E}_2$, as shown by the filled in blue circles in Figure 2.7 (iv, v).

The red point in Figure 2.7 (iv) shows the result of the first correction step, where the ESPs $\left([\text{Enz}]_0, k_c\right)$ were fixed to one of the points estimated in the calibration step (red point on the ESP plane), and the CSP was estimated. We see that the CSP value estimated is such that the full parameter point lies in the joint ESP-CSP set (red point lifted up to the green surface). The fitted trajectories from this stage are shown in Figure 2.7 (vii).

We observe from the position of the red point in Figure 2.7 (v) that picking an arbitrary point from the set of ESPs, and using the CSPs from the first correction step leads to a point that does not lie on the joint ESP-CSP surface for extract $\mathcal{E}_1$. The corresponding predicted correction and the true behavior of the artificial data are shown in Figure 2.7 (viii).

Figure 2.7 (vi, ix) show the result of repeating the procedure with the CSP fixing modification applied at the calibration step. In particular, the CSP was fixed at the value that was estimated at the first calibration step (lifted red point in Figure 2.7 (iv)), though any value in the set $\text{proj}_c\, \Theta_{1,\text{cal}} \cap \text{proj}_c\, \Theta_{2,\text{cal}}$ is allowed. The key insight here is that now the ESP sets are much smaller, and in correction step one, the ESPs can only be picked so that the very CSP value that was fixed gets estimated, and subsequently, in correction step two, the only ESP values that can be picked are such that when they are used with this CSP value, the resulting point lies in the set of parameters $\Theta_{1,\text{cal}}$ that fit the true $\mathcal{E}_1$ data to the model. In Figure 2.7 (ix), we see that this leads to the desired correction.

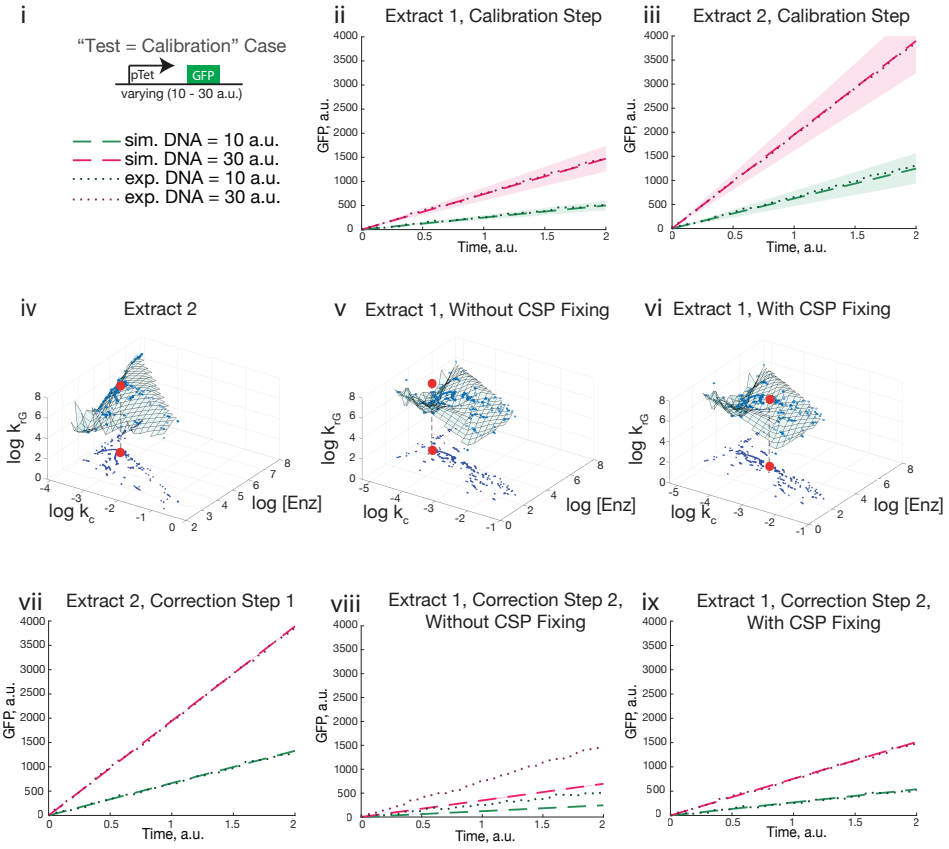

Figure 2.7: *In silico* (model universe) demonstration of the effect of thin covariation and CSP fixing for the 'Test = Calib' case of the data correction problem. (i) Dotted lines: Artificial experimental data, with a small amount of Gaussian noise added for easier visualization (fits were performed on the noise free trajectories). The data was generated using the constitutive expression model of Equation (2.8) at DNA concentrations of 10 and 30 arbitrary units (a.u.). Dashed lines and shaded regions: means and standard deviations of simulated trajectories using parameter points drawn from the estimated posterior distributions. (ii - iii) Artificial data generated using known parameters for two extracts. The CSPs used were the same for the models in both extracts $\bar{c} = (\bar{k}_{fG}, \bar{k}_{rG}) = (5, 300)$, while the ESPs differed for the two extracts, $\bar{e}_1 = (\bar{k}_{c1}, \overline{\mathrm{Enz}}_{0,1}) = (0.012, 100)$, $\bar{e}_2 = (\bar{k}_{c2}, \overline{\mathrm{Enz}}_{0,2}) = (0.024, 200)$. The model fits to the data are the dashed lines and shaded regions, and the parameter distribution for the three parameters estimated ($k_{fG} = 5$ fixed) is shown as the blue empty circles and the fitted translucent green surface in (vi, v). We note that this model shows thin covariation as is visible from the two dimensional surface fit (embedded in 3D) to the scattering of points. Solid blue circles are the projection of the parameter points onto the ESP subspace. (continued below)

Figure 2.7: (continued from above) The plot (iv) also depicts the first correction step, with the red point on the $k_c - [\mathrm{Enz}]_0$ plane denoting the point $\hat{e}_{2,\mathrm{cal}} \in E_{2,calib}$ and the red point on the surface showing the corresponding estimated CSPs. (vii) The fits corresponding to the CSP estimation of correction step one. (v) Correction step two showing an arbitrary ESP point $\hat{e}_1 \in E_{1,\mathrm{cal}}$ with the estimated CSP from (iv) leading to a point that is off the surface of points denoting the set of all parameters that fit extract 1 data to the model. (viii) Corresponding corrections fail. (vi) Correction step two with CSP fixing at the calibration step, and the corresponding corrected trajectories (ix).

2.7.2 Application of CSP Fixing in the General Setting

We conclude this section by demonstrating that when CSP fixing is used in the general case when the test circuit is not the same as the calibration circuit, the CSP fixing modification to the method still leads to significant improvements in the performance of the method (Figure 2.8). The calibration data used was the same as in Section 2.7.1, and the test circuit model (Figure 2.8 (i)) used was the one in Equations (2.9), with parameters used to generate the artificial data given in Table 2.1. As before, dotted lines denote artificial data with a small amount of noise added for ease of visualization only (all the fitting was done on noise free data). The calibration stage with and without CSP fixing was identical to that in Section 2.7.1. 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},$k_{f,dim}$, and $k_{f,rep}$, and limited the CSPs to only the reverse rates, k_{rG}, k_{rT},$k_{r,dim}$, and $k_{r,rep}$. In this setting, performing the first correction step gave a set of parameter estimates for the CSPs, and the resulting fits to the $\mathcal{E}_2$ test circuit data are shown in Figure 2.8 (ii). Performing the second correction step led incorrect prediction of the corrected trajectories (failure condition two), as shown in Figure 2.8 (iii). Significantly, applying the CSP fixing modification to the calibration step led to good prediction of the circuit in $\mathcal{E}_2$, as shown in Figure 2.8 (iv).

2.8 Discussion and Future Work

Cell-free extract *in vitro* systems are becoming a useful prototyping tool in synthetic biology, yet the intrinsic variability between the batches of these extracts places limitations on the comparability of results obtained in different batches. Indeed, users currently plan

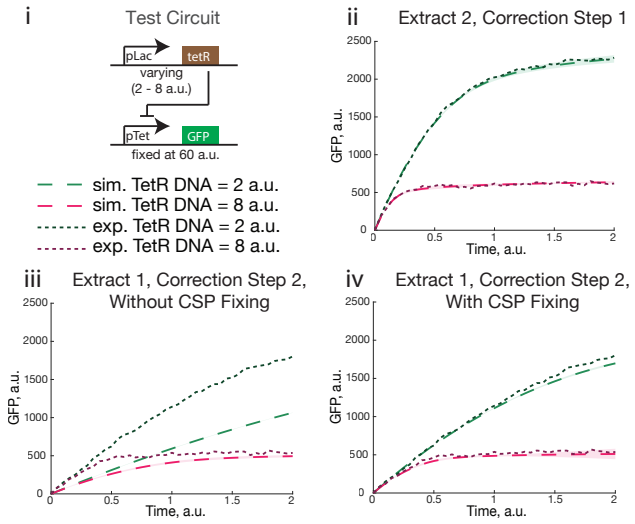

Figure 2.8: The effect of CSP fixing at the calibration step on the correction of novel test circuit data. (i) The test circuit was the repression of the pTet promoter, modeled by Equations (2.9). The pTet-GFP DNA was held fixed at 60 a.u., while the constitutive tetR DNA was varied between 2 a.u. and 8 a.u.. The dotted lines were the artificial experimental data generated using the parameters in Table 2.1. The calibration step was performed as in Figure 2.7, both with and without CSP fixing. The first correction step leads to the fits shown in (ii), and the second correction step leads to the poor corrections shown in (iii). When CSP fixing is employed at the calibration step, the second correction step performs well, as shown in (iv).

their investigations so that all their experiments may be completed before the batch of extract runs out. For this reason, they are limited in the number of things they are able to compare under identical experimental conditions.

We have demonstrated a model-based methodology for calibrating extracts that allows for genetic circuit behavior to be normalized or *corrected*. This methodology is organized into two steps, a *calibration step*, where a set of calibration circuits is used to estimate extract specific parameters of a particular extract, and a *correction step*, in which the calibrations are used to transform a novel circuit's behavior from what it was in a given extract into what it would have been in a *reference* extract. The general 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 extract 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 transformed into its reference extract form.

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

Correcting for behavior between topologically different environments, which may arise when *in vitro*-to-*in vivo* prediction of behavior is attempted, or when correcting for variability between different bacterial strains is required, may also be achieved if this method is generalized in a manner outlined next. Briefly stated, the method would still assume that as long as the modeling framework and environment specific parameters are chosen well enough to capture most of the environment specific influences on the circuit (in each environment), then the circuit specific parameters should be largely independent of the environment they are estimated in. This should allow for an *environment specific* set of

calibration experiments to be designed and used in the calibration step, followed by the first and second correction steps that are largely similar to those in the methodology outlined here. The appropriateness of the choice of the level of detail in the modeling, the partition of parameters into extract specific versus circuit specific and the choice of calibration experiments in each of the different environments may be achieved in an iterative, empirical, hypothesis driven manner.

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 species that can be measured as outputs, parameter non-identifiability is a ubiquitous property of these models. The general prescription in modeling studies [2] is to perform a greater number of experiments to eliminate non-identifiability, reduce the order of the model to reduce the number of parameters, 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 for independent reasons (one example being the explicit modeling of nucleotide binding and consumption during transcription and translation to keep track of resources). The fixing of some parameters, while reducing the number of effective parameters may not remove non-identifiability completely.

The main insight behind our theoretical results is that since we are only trying to correct the trajectories of the very species that we are able to measure in the first place, perhaps the sets of values the non-identifiable parameters can take can be treated as equivalence classes with respect to their usage in our modeling framework. This is a general idea that, even though developed and demonstrated in this specific framework, should apply to a broader class of applications of parametric models, as long as those applications depend on using only the observable outputs. A 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.

We can identify a few other directions of investigation for future work. Firstly, condition 2.14 in Theorem 1 might be generalizable to a similar result which gives conditions under which part models with parameter non-identifiability can be combined to predict the behavior of an entire system. In the simplest case, this could be a simple Cartesian product condition, though we suspect 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, before the remaining non-identifiability does not matter for the output prediction problem. Secondly, we believe that it should be possible to use the result from the theory of differential equations that specifies the continuous dependence of model outputs on parameters to show that the direction of movement, when the outputs are varied under a fixed set of experimental conditions, of a non-identifiable parameter set must be orthogonal to the direction of the non-identifiability, and indeed the non-identifiability must be 'thin', in some geometric sense, in the direction of movement. Lastly, 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 [53] are incorporated into our analysis.

Appendices

2.A 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 6 and Remark 4.

Lemma 2. *Let* $\tilde{\Theta}_{\text{cal}}$, $\Theta_{1,\text{cal}}$ *and* $\Theta_{2,\text{cal}}$ *be as defined in Definition 6 and Remark 4. Then, the identities*

$$\text{proj}_c\, \tilde{\Theta}_{\text{cal}} \equiv \text{proj}_c\, \Theta_{1,\text{cal}} \cap \text{proj}_c\, \Theta_{2,\text{cal}}, \tag{2.26}$$

$$\text{proj}_{e_i}\, \tilde{\Theta}_{\text{cal}} \equiv \left\{ e \;\middle|\; \exists c \in \left(\text{proj}_c\, \Theta_{1,\text{cal}} \cap \text{proj}_c\, \Theta_{2,\text{cal}} \right) : (e, c) \in \Theta_{i,\text{cal}} \right\}, \qquad i = 1,\, 2, \tag{2.27}$$

hold.

Proof. First, we prove (2.26) using a series of equivalences. Let $\tilde{c} \in \text{proj}_c\, \tilde{\Theta}_{\text{cal}}$. This is equivalent to

$$\exists e_1, e_2 \,:\, (e_1, e_2, \tilde{c}) \in \tilde{\Theta}_{\text{cal}} \tag{2.28}$$

$$\Leftrightarrow \exists e_1, e_2 \,:\, \overline{y}_{i,\text{cal}} = M_{\text{cal}}(e_i, \tilde{c}), \qquad i = 1, 2 \tag{2.29}$$

$$\Leftrightarrow (e_i, \tilde{c}) \in \Theta_{i,\text{cal}}, \qquad i = 1, 2 \tag{2.30}$$

$$\Leftrightarrow \tilde{c} \in \text{proj}_c\, \Theta_{1,\text{cal}} \cap \text{proj}_c\, \Theta_{2,\text{cal}}, \tag{2.31}$$

which proves the assertion.

Next, we prove (2.27) for e_1 by showing that the left and right hand sides are subsets of each other. The proof for the e_2 case is similar. Denote the set on the left hand side with L, and the one on the right with R. Let $\tilde{e}_1 \in L = \text{proj}_{e_1}\, \tilde{\Theta}_{\text{cal}}$. Then, $\exists \tilde{e}_2, \tilde{c}$ such that

$(\tilde{e}_1, \tilde{e}_2, \tilde{c}) \in \tilde{\Theta}_{\text{cal}}$, which implies $\tilde{c} \in \text{proj}_c \tilde{\Theta}_{\text{cal}}$ and $\overline{y}_{1,\text{cal}} = M_{\text{cal}}(\tilde{e}_1, \tilde{c})$. By the identity (2.26), we have that $\tilde{c} \in \text{proj}_c \Theta_{1,\text{cal}} \cap \text{proj}_c \Theta_{2,\text{cal}}$ and $(\tilde{e}_1, \tilde{c}) \in \Theta_{1,\text{cal}}$, which shows that $L \subseteq R$.

We conclude the proof by showing that $R \subseteq L$. Let $\tilde{e}_1 \in R$, which means that there exists a $\tilde{c} \in \text{proj}_c \Theta_{1,\text{cal}} \cap \text{proj}_c \Theta_{2,\text{cal}}$ such that $\overline{y}_{1,\text{cal}} = M_{\text{cal}}(\tilde{e}_1, \tilde{c})$. Furthermore, since $\tilde{c} \in \text{proj}_c \Theta_{2,\text{cal}}$, there also exists an $\tilde{e}_2$ such that $\overline{y}_{2,\text{cal}} = M_{\text{cal}}(\tilde{e}_2, \tilde{c})$. Together these imply that $(\tilde{e}_1, \tilde{e}_2, \tilde{c}) \in \tilde{\Theta}_{\text{cal}}$, which gives $\tilde{e}_1 \in \text{proj}_{e_1} \tilde{\Theta}_{\text{cal}}$, proving the assertion. □

2.B Equivalence of the Two CSP Subset Conditions Given in Remark 7

The Cartesian product condition given in Equation (2.14) implies two further conditions, which we state in Lemma 3 below. The first of these follows simply by projecting both sides of Equation (2.14) onto the ESP coordinates. The second condition, on the other hand, is stronger than simply projecting (2.14) onto the CSP coordinates. This condition states that the CSP points generated at the first correction step, $C'_{2,\text{test}}$, must be a subset of the set of CSP points generated by fitting $\overline{y}_{1,\text{test}}$ to the model when the ESP points are restricted to be in the set $E_{1,\text{cal}}$.

Lemma 3. *Condition* (2.14)*, which states that* $E_{1,\text{cal}} \times C'_{2,\text{test}} \subseteq \Theta_{1,\text{test}}$*, implies that*

$$E_{1,\text{cal}} \subseteq \text{proj}_e \Theta_{1,\text{test}}, \tag{2.32}$$

$$C'_{2,\text{test}} \subseteq C'_{1,\text{test}}, \tag{2.33}$$

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

$$C'_{1,\text{test}} \triangleq \bigcup_{\hat{e} \in E_{1,\text{cal}}} \text{ID}_{c|e=\hat{e}}\left(\overline{y}_{1,\text{test}}, M_{\text{test}}(e, c)\right).$$

Proof. Condition (2.32) follows simply by applying the proj_e operator to both sides of condition (2.14). To prove condition (2.33), we note that condition (2.14) implies that for an arbitrary $\tilde{c} \in C'_{2,\text{test}}$, we have that for all $\tilde{e} \in E_{1,\text{cal}}$, the model fits the data, $\overline{y}_{1,\text{test}} = M_{\text{test}}(\tilde{e}, \tilde{c})$.

This in turn implies that

$$\tilde{c} \in \bigcup_{\hat{e} \in E_{1,\text{cal}}} \text{ID}_{c|e=\hat{e}} \left(\overline{y}_{1,\text{test}}, M_{\text{test}}(e, c) \right) = C'_{1,\text{test}}. \tag{2.34}$$

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