Online Model Selection for Synthetic Gene Networks

Wei Pan, Filippo Menolascina and Guy-Bart Stan

***Abstract*— Control algorithms combined with microfluidic devices and microscopy have enabled *in vivo* real-time control of protein expression in synthetic gene networks. Most control algorithms rely on the *a priori* availability of mathematical models of the gene networks to be controlled. These models are typically black/grey box models, which can be obtained through the use of data-driven techniques developed in the context of systems identification. Data-driven inference of both model structure and parameters is the main focus of this paper. There are two main challenges associated with the inference of dynamical models for real-time control of gene regulatory networks in living cells. Since biological systems are typically evolving over time, the first challenge stems from the fact that model selection needs to be done online, which prevents the application of computationally expensive identification algorithms iterating through large amounts of streaming data. The second challenge consists in performing nonlinear model selection, which is typically too burdensome for Kalman filtering related techniques due the heterogeneity and nonlinearity of the candidate models. In this paper, we combine sparse Bayesian techniques with classic Kalman filtering techniques to tackle these challenges.**

I. Introduction

The problem of identifying biological networks from experimental time series is of fundamental interest in systems and synthetic biology. For example, such information can aid in the design of drugs or of synthetic biology genetic controllers [1]. Methods developed in the context of system identification [2] can be applied for such purposes. While predictive models proved to play a key role in the control of synthetic gene circuit [3], [4], highly detailed or complex models are typically difficult to obtain, analyse and control. Therefore, one typically prefers to use simple or sparse models that capture at best the dynamics expressed in the collected data. The identification and use of simple or sparse models inevitably introduces uncertainties in both the structure and the parameters of the models [5], [6]. To reduce the impact of such uncertainties on our understanding of the underlying processes, it is common practice in quantitative biology to use multiple experimental replicates to study a network of interest, generally represented by a non-linear dynamical system.

The problem of identifying the parameters and/or the states of a non-linear system is widespread in control systems engineering and received much attention in the recent history: from Extended (or Unscented) Kalman Filters to Particle Filters and Markov Chain Monte Carlo methods, many techniques have been developed over the past decades to address such problems. However these methods all require some prior knowledge of the model structure, a luxury seldom met in the context of synthetic gene circuits. What is more, when the structure of the model is itself uncertain these methods typically perform parameter estimation for all the candidate models, in a sort of "brute-force approach". All the identified models are then compared against each other using some information criterion (e.g. AIC, BIC, see [2], [7]).

To improve on this practice we present here a framework for efficiently identifying, from time-series data, nonlinear ordinary differential equations models, with a special emphasis on functional forms or dependencies, that are commonly used to capture the dynamics of gene regulatory networks.

Notation: The notation in this paper is standard. Bold symbols are used to denote vectors and matrices. For a matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$, $\mathbf{A}(i,j) \in \mathbb{R}$ denotes the element in the i^{th} row and j^{th} column, $\mathbf{A}(i,:) \in \mathbb{R}^{1\times N}$ denotes its i^{th} row, $\mathbf{A}(:,j) \in \mathbb{R}^{M\times 1}$ denotes its j^{th} column. For a column vector $\boldsymbol{\alpha} \in \mathbb{R}^{N\times 1}$, α_i denotes its i^{th} element. In particular, $\mathbf{I}_L$ denotes the identity matrix of size $L\times L$. We simply use $\mathbf{I}$ when the dimension is obvious from context. $\|\boldsymbol{\beta}\|_1$ denotes the ℓ_1 norm of the vector $\boldsymbol{\beta}$. $\mathrm{diag}\,[\gamma_1,\ldots,\gamma_N]$ denotes a diagonal matrix with principal diagonal elements being $\gamma_1,\ldots,\gamma_N$. $\mathrm{blkdiag}[\mathbf{A}^{[1]},\ldots,\mathbf{A}^{[C]}]$ denotes a block diagonal matrix with principal diagonal blocks being $\mathbf{A}^{[1]},\ldots,\mathbf{A}^{[C]}$. $\mathrm{Tr}(\mathbf{A})$ denotes the trace of $\mathbf{A}$. A matrix $\mathbf{A} \succeq \mathbf{0}$ means $\mathbf{A}$ is positive semidefinite. A vector $\boldsymbol{\gamma} \succeq \mathbf{0}$ means each element in $\boldsymbol{\gamma}$ is nonnegative.

II. Related Work

A. Background

Transcription and translation are two intrinsically slow processes (time scale of minutes in bacteria). While, on one hand, this implies there is no stringent need to observe cells with high sampling frequencies, it also means that identification/control experiments of biomolecular circuits usually last hours to days, i.e. much longer than similar experiments carried out on electrical or mechanical systems. Over such long time frames it is necessary to **(a)** effectively trap cells and **(b)** observe their internal dynamics. We also need to extract single cell trajectories while we **(c)** stimulate them with time-varying profiles of the molecules that serve

Dr Wei Pan gratefully acknowledges the support of Microsoft Research through the PhD Scholarship Program and EPSRC Centre for Mathematics of Precision Healthcare (EP/N014529/1). Dr Guy-Bart Stan gratefully acknowledges the support of the EPSRC Fellowship for Growth (project EP/M002187/1). Dr Filippo Menolascina gratefully acknowledges the support of the Wellcome Trust-University of Edinburgh Institutional Strategic Support Fund.

W. Pan is with the Data Science Institute and Department of Mathematics, Imperial College London, United Kingdom. Email: w.pan11@imperial.ac.uk

F. Menolascina is with the Institute for Bioengineering, School of Engineering, University of Edinburgh, United Kingdom. Email: filippo.menolascina@ed.ac.uk

G.-B. Stan is with the Department of Bioengineering, Imperial College London, United Kingdom. Email: g.stan@imperial.ac.uk

as inducers for the network of interest. Most importantly it is necessary to achieve these objectives with minimally invasive techniques, i.e. using methods that ideally, will not affect the processes we want to quantify (a point not to be overlooked as factors like heat, e.g. generated by the light used to obtain microscopy images, or mechanical stresses, e.g. used to physically hold cells in place while imaging them, will trigger stress responses in cells). For these reasons we need to continuously **(i)** supply cells with nutrients and **(ii)** remove toxic metabolites while **(iii)** retaining the ability to condition their microenvironment to expose them to the appropriate externally applied stimuli. All these requirements, combined, significantly limit the technologies that can be used to identify and control biomolecular circuits ***in vivo***: for example commonly used methods, such as flask-based sampling or bioreactors, are unable to provide us with single cell trajectories. Microfluidics, enabling us to fabricate transparent microchannels where cells can be trapped and observed while being exposed to a continuous flow of nutrients and chemicals controlled by a computer, does allow to meet the requirements mentioned above.

In the view of the above, we will consider the setup documented in [3] as the reference platform for the ***in vivo*** implementation of our model selection approach. In this configuration, described in Fig 1, a microfluidic device containing the cells carrying the network of interest, is mounted on the stage of a fully automated microscope that takes phase contrast and fluorescence images of the cells at regular time intervals. Such images are used by the computer to locate cells (phase contrast) and estimate the amount of protein (fluorescence imaging) in real-time via a custom image processing algorithm developed in MATLAB. The computer, then, uses a set of fluidic pressure actuators to vary the level of inducer the cells are exposed to. Interestingly, this configuration allows us to continuously **(a)** update our model on-line and, potentially, **(b)** automtically carry out multiple model-selection iterations within the same experiment, a unique feature of this approach [8].

In order to extend the experimental throughput and increase our model discrimination capabilities we will use the MDAW microfluidic device described in [9]: in this device 8 independent model selection experiments can be carried out at the same time. We will seed the same strain in each of the 8 chambers and image the 8 chambers at regular intervals. In so doing we will obtain 8 independent datasets (each formed by an exponentially growing number of single cell trajectories) that we will use to design and implement our model selection experiment.

1) Mathematical model: Throughout the paper, we will assume that the process of interest can be modelled by a discrete-time system of the form:

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{g}(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\beta}) + \boldsymbol{\upsilon}_t, \\ \mathbf{z}_t &= \mathbf{x}_t + \boldsymbol{\eta}_t, \end{aligned} \tag{1}$$

where the $\mathbf{x}_t = [x_{1,t}, \ldots, x_{n_\mathbf{x},t}] \in \mathbb{R}^{n_\mathbf{x}}$ is the state vector at discrete time point t; $\boldsymbol{\beta}$ represents the vector of parameters to be identified; the function $\mathbf{g} : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \times \mathbb{R}^{n_\beta} \to \mathbb{R}^{n_\mathbf{x}}$ is nonlinear and depends (explicitly) on the input vector $\mathbf{u}_t \in \mathbb{R}^{n_\mathbf{u}}$. The process noise $\boldsymbol{\upsilon}_t \in \mathbb{R}^{n_\mathbf{x}}$ and measurement noise $\boldsymbol{\eta}_t \in \mathbb{R}^{n_\mathbf{x}}$ are assumed to be mutually independent Gaussian random variables with known positive covariance matrices $\mathbf{Q}_t$ and $\mathbf{R}_t$, respectively.

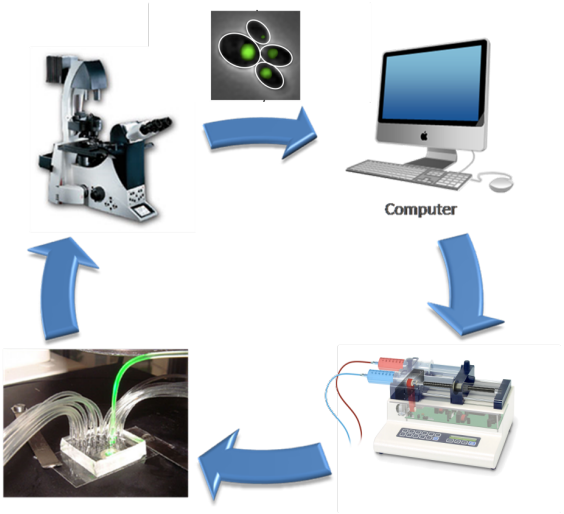


Fig. 1. **Technological platform for *in-vivo* model selection of synthetic circuits.** In this closed loop configuration the computer (upper right corner) takes images of the cells in the microfluidic device (lower left corner) via a microscope (upper left corner), quantifies the output of the network of interest in real time and applies the next sample of input(s) via the fluidic pressure actuation system (lower right corner).

The state vector $\mathbf{x}_t$ usually contains concentrations of certain chemical species of interest, such as mRNAs or proteins. The output signal $\mathbf{z}_t$ represents the quantities we can measure experimentally.

B. Questions of interest

1) Estimation of the model structure, i.e. the functional structure of $\mathbf{g}_n(\cdot)$ in (1).
2) Estimation the parameter vector $\boldsymbol{\beta}$ therein.
3) Identification of a single model from multiple datasets emanating from perturbation experiments performed on systems of the form given in (1) that differ in terms of their parameters but not their parametric structure.

For example, in the ideal noiseless case, a simple self-induction gene network can be described as [10]:

$$\begin{aligned} \frac{dx_t}{dt} &= -kx_t + \frac{V_{\max} x_t^h}{K_M + x_t^h}, \\ z_t &= x_t. \end{aligned} \tag{2}$$

where $\frac{dx_t}{dt}$ is a numerical estimation of the time derivative of x_t (see appendix of [11] for details), k is the decay rate of gene product x, $V_{\max}$ is the maximum gene expression rate, K_M is the threshold value in terms of the concentration of gene product x that results in a production rate of $0.5V_{\max}$, and h is the Hill coefficient (a.k.a. the cooperativity coefficient) associated with the self-induction of gene x.

The identification problem can be formulated as: given some time series data corresponding to discrete time point measurements of the gene product, i.e. $z_1, z_2, \cdots, z_{M+1}$, can the model given in (2) be identified or approximated?

III. Model Structure Identification

A. Addressing questions of interest 1 and 2 by solving a linear regression problem

In our previous work [12], [11], we developed a framework for nonlinear ODE model identification based on the following assumption:

Assumption 1: The nonlinear functions that define the ODE model are expressed as a linear combination of nonlinear terms which do not contain unknown paramaters. The only unknown parameters are the linear coefficient that define the linear combination.

When this assumption is satisfied, the system in (1) can be expressed as follows for $n = 1, \ldots, n_{\mathbf{x}}$:

$$\begin{aligned} x_{n,t+1} &= \mathbf{g}_n(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\beta}) + \boldsymbol{\upsilon}_{n,t} && (3)\\ &= \sum_{s=1}^{N_n} \beta_{ns} f_{ns}(\mathbf{x}_t, \mathbf{u}_t) + \boldsymbol{\upsilon}_{n,t}, && (4)\\ z_{n,t} &= x_{n,t} + \boldsymbol{\eta}_{n,t}, && (5) \end{aligned}$$

where $\beta_{ns} \in \mathbb{R}$ and $\{f_{ns}(\mathbf{x}_t, \mathbf{u}_t) : \mathbb{R}^{n_{\mathbf{x}}} \times \mathbb{R}^{n_{\mathbf{u}}} \to \mathbb{R}, s = 1, \ldots, N_n\}$ defines the set of *all* candidate/possible basis functions that govern the dynamics of x_n. The functions $f_{ns}(\mathbf{x}_t, \mathbf{u}_t)$ are assumed to be Lipschitz continuous.

Suppose that the right-hand side of equation (4) is unknown. We showed in [12] that, under Assumption 1, we can address the first two questions of interest within the frame box in Section II-B by identifying the unknown parameters in the following linear regression problem

$$\mathbf{y}_n = \mathbf{A}_n \boldsymbol{\beta}_n + \boldsymbol{\xi}_n, \ n = 1, \ldots, n_{\mathbf{x}} \tag{6}$$

Suppose the data are collected at $M+1$ time instances and N_n basis functions are used in the expansion given in (4). Equation (6) then has the following structure:

$$\begin{aligned} &\mathbf{y}_n \triangleq [z_{n,2}, \ldots, z_{n,M+1}]^\top \in \mathbb{R}^M\\ &\mathbf{A}_n \triangleq [\mathbf{A}_n(:,1), \ldots, \mathbf{A}_n(:,N_n)]\\ &\quad = \begin{bmatrix} f_{n1}(\mathbf{z}_1, \mathbf{u}_1) & \ldots & f_{nN_n}(\mathbf{z}_1, \mathbf{u}_1) \\ \vdots & & \vdots \\ f_{n1}(\mathbf{z}_M, \mathbf{u}_M) & \ldots & f_{nN_n}(\mathbf{z}_M, \mathbf{u}_M) \end{bmatrix} \in \mathbb{R}^{M \times N_n},\\ &\boldsymbol{\beta}_n \triangleq [\beta_{n1}, \ldots, \beta_{nN_n}]^\top \in \mathbb{R}^{N_n}\\ &\boldsymbol{\xi}_n \triangleq [\xi_{n1}, \ldots, \xi_{nM}]^\top \in \mathbb{R}^M \end{aligned} \tag{7}$$

The noise vector $\boldsymbol{\xi}_n$ in (6) is still Gaussian distributed with zero mean but is now characterised by a non-diagonal (possibly fully-parametrised) covariance matrix $\boldsymbol{\Pi}_n \in \mathbb{R}_+^{M \times M}$ (see Appendix of [11]). The solution $\boldsymbol{\beta}_n$ to the linear regression problem in (6) is typically going to be sparse, which is mainly due to the potential introduction of non-relevant and/or non-independent dictionary functions in $\mathbf{A}_n$.

Since the $n_{\mathbf{x}}$ linear regression problems in (6) are independent, for simplicity of notation, we omit the subscript n used to index the state variable and simply write Eq. (6) as:

$$\mathbf{y} = \mathbf{A}\boldsymbol{\beta} + \boldsymbol{\xi}, \tag{8}$$

B. Addressing questions of interest 1, 2 and 3 by performing identification from multiple datasets

In this section, we will show how the third point in Section II-B can be addressed.

To ensure reproducibility, experimentalists repeat their experiments under the same conditions, and the collected data are then called "replicates". Typically, only the average value over these replicates is used for modelling or identification purposes. In this case, however, only the first moment is used and information provided by higher order moments is lost. Moreover, when data originate from different experimental conditions, it is usually very hard to combine the datasets into a single identification problem. This section will address these issues by showing how several datasets can be combined to define a unified optimisation problem whose solution is an identified model consistent with the various datasets available for identification. This can be done using the approach proposed in [11], which consists in merging a total number of C datasets collected from C independent experiments. We put a subscript $[c]$ to index the identification problem associated with the specific dataset obtained from experiment $[c]$, i.e. we replace $\mathbf{A}$ with $\mathbf{A}^{[c]}$, and similarly for $\mathbf{y}$, $\boldsymbol{\beta}$ and $\boldsymbol{\xi}$. The linear regression problem in (6) can then be written as:

$$\mathbf{y}^{[c]} = \mathbf{A}^{[c]}\boldsymbol{\beta}^{[c]} + \boldsymbol{\xi}^{[c]}, \quad c = 1, \ldots, C. \tag{9}$$

Let $\mathbf{A}_i = \text{blkdiag}[\mathbf{A}^{[1]}(:,i), \ldots, \mathbf{A}^{[C]}(:,i)]$, and $\boldsymbol{\beta}_i = [\beta_i^{[1]}, \ldots, \beta_i^{[C]}]^\top$, for $i = 1, \ldots, N$. We further define

$$\begin{aligned} \mathbf{y} &= \begin{bmatrix} \mathbf{y}^{[1]} \\ \vdots \\ \mathbf{y}^{[C]} \end{bmatrix}, \mathbf{A} = \begin{bmatrix} \ \mathbf{A}_1 \ \big| \ \cdots \ \big| \ \mathbf{A}_N \ \end{bmatrix},\\ \boldsymbol{\beta} &= \left[\begin{array}{c} \boldsymbol{\beta}_1 \\ \hline \vdots \\ \hline \boldsymbol{\beta}_N \end{array}\right], \boldsymbol{\xi} = \left[\begin{array}{c} \boldsymbol{\xi}^{[1]} \\ \hline \vdots \\ \hline \boldsymbol{\xi}^{[C]} \end{array}\right], \end{aligned} \tag{10}$$

which gives

$$\mathbf{y} = \mathbf{A}\boldsymbol{\beta} + \boldsymbol{\xi}. \tag{11}$$

This yields a formulation very similar to that presented previously in (8). However, in the multi-experiment formulation (11), there is now a special block structure for $\mathbf{y}$, $\mathbf{A}$ and $\boldsymbol{\beta}$. Note also that the experimental platform described in Section II-A specifically allows multiple experiments to be carried out at the same time, therefore exploiting the power of the approach we describe. Further detail can be found in [11].

In what follows, we give a self-contained description of the corresponding identification algorithm. The reader is referred to [11] for a full explanation of the algorithm and of its variables.

C. Challenges inherent to the identification of biological and biochemical systems

The *a priori* selection of a good set of dictionary functions $f_{ns}(\mathbf{x}, \mathbf{u})$ in (4) is key to the identification process. Some *a priori* knowledge of the provenance of the data and the

Algorithm 1 Identification Algorithm using Heterogeneous Datasets

1: Collect C groups of time series data from C independent experiments;
2: Select the candidate basis functions that will be used to construct the dictionary matrix described in Section III;
3: Initialise $\theta_i^0 = 1,\ \forall i,\ \alpha_i^0 = \frac{\theta_i^0}{C},\ \Lambda^0 = \mathbf{I}$;
4: Initialise $\mathbf{S}^0 = \frac{1}{\lambda}\mathbf{I},\ \lambda = 1$;
5: **for** $k = 0, \ldots, k_{\max}$ **do**
6: $\boldsymbol{\beta}^{k+1}$ can be obtained by solving:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\left(\mathbf{y} - \mathbf{A}\boldsymbol{\beta}\right)^\top \mathbf{S}^k \left(\mathbf{y} - \mathbf{A}\boldsymbol{\beta}\right) + \sum_{i=1}^{N} \|\theta_i^k \cdot \boldsymbol{\beta}_i\|_2; \tag{12}$$

7: Update $\gamma_i^{k+1} = \frac{\|\boldsymbol{\beta}_i^{k+1}\|_2}{\sqrt{C\alpha_i^k}}$.
8: Let $\mathbf{Y}^{k+1} = (\mathbf{A}\boldsymbol{\beta}^{k+1} - \mathbf{y}) \cdot (\mathbf{A}\boldsymbol{\beta}^{k+1} - \mathbf{y})^\top$;
9: $\mathbf{S}^{k+1}$ can be obtained by solving:

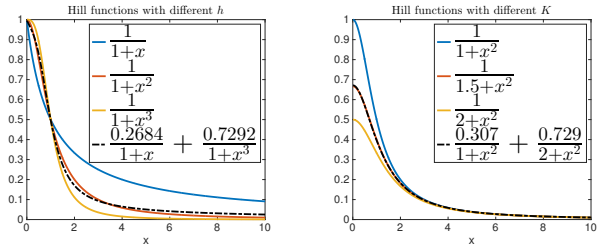
$$\min_{\mathbf{S} \succeq \mathbf{0}} \operatorname{Tr}\left(\mathbf{Y}^{k+1} + \Lambda^k\right)\mathbf{S} - \log\det\mathbf{S}; \tag{13}$$

10: Update $\boldsymbol{\alpha}^{k+1} = \operatorname{diag}\{\left[-(\boldsymbol{\Gamma}^k)^{-1} + \mathbf{A}^\top\mathbf{S}^k\mathbf{A}\right]^{-1}\} \cdot \operatorname{diag}\{-(\boldsymbol{\Gamma}^k)^{-2}\} + \operatorname{diag}^{-1}\{\boldsymbol{\Gamma}^k\}$;
11: Update $\theta_i^{k+1} = C\alpha_i^{k+1}$;
12: Update $\Lambda^{k+1} = \mathbf{A}(\boldsymbol{\Gamma}^{-k} + \mathbf{A}^\top\mathbf{S}^k\mathbf{A})^{-1}\mathbf{A}^\top$;
13: **if** a stopping criterion is satisfied **then**
14: Break;
15: **end if**
16: **end for**

field for which the models are developed can be particularly helpful for this. For example, the typical nonlinearities used to create nonlinear ODE models of gene regulatory networks can be restricted to those known to capture fundamental biochemical kinetic laws, e.g. first-order functions $f(x) = \alpha x$, mass action functions $f([x_1, x_2]) = \beta x_1 \cdot x_2$, Michaelis-Menten functions $f(x) = \frac{V_{\max}}{K+x}$, or Hill functions $f(x) = \frac{V_{\max}}{K+x^h}$. Using our framework, as stated in Assumption 1, h and K are assumed to be known *a priori*, whereas $\alpha, \beta, V_{\max}$ can be identified through the process described in the previous sections.

A first challenge using our framework is thus to find practical solutions to the identification of the parameters embedded nonlinearly in the dictionary functions, e.g. the paramters h and K of the Hill functions.

A naive solution to the estimation of the Hill coefficient, h, is to introduce more nonlinear terms in the set of dictionary functions, each with a different Hill coefficient. Since $h \in \mathbb{Z}^+$ and very few biological systems are characterised by Hill coefficients larger than 8, the number of such terms is typically relatively low. On the basis of this, the set of Hill functions $\frac{V_{\max}}{K+x^h}$ with $h = 1, 2, \cdots, 8$ is a good candidate subset to be included in the set of dictionary functions. Furthermore, even if the true function is not a member of the considered set of dictionary functions, it is often the case that the true dictionary function can be approximated by a linear combination of the other members



(a) Hill functions $\frac{1}{1+x^h}$ characterised by different Hill coefficients, h, with $h = \{1, 2, 3\}$. The Hill function $\frac{1}{1+x^2}$ is tightly approximated by the linear combination $\frac{0.2684}{1+x} + \frac{0.7292}{1+x^3}$.

(b) Hill functions $\frac{1}{K+x^2}$ characterised by different Hill thresholds, K, with $K = \{1, 1.5, 2\}$. The Hill function $\frac{1}{1.5+x^2}$ is tightly approximated by the linear combination $\frac{0.307}{1+x^2} + \frac{0.729}{2+x^2}$.

Fig. 2. Hill functions can be approximated by linear combinations of other Hill functions.

of the set of dictionary functions. For example, suppose the true function to be identified is $\frac{1}{1+x^2}$ and that the set of dictionary functions is $\{\frac{1}{1+x}, \frac{1}{1+x^3}, \frac{1}{1+x^4}\}$. The true function $\frac{1}{1+x^2}$ can be approximated as a linear combination of the other rational functions present in the set of dictionary functions: $\frac{1}{1+x^2} \approx a \cdot \frac{1}{1+x} + b \cdot \frac{1}{1+x^3} + 0 \cdot \frac{1}{1+x^4}$, where a and b are some real numbers that can be identified using our framework, see Figure 2(a).

The estimation of the Hill threshold parameter K can be dealt with in a similar manner as for the Hill coefficient h. For example, the nonlinear Hill function $\frac{1}{1.5+x^2}$ can be approximated by a linear combination of Hill functions with different values of K: $\frac{1}{1.5+x^2} \approx \frac{a}{1+x^2} + \frac{b}{2+x^2}$, where a and b are some real numbers that can be identified using our framework, see Figure 2(b).

Another important challenge emerges when one wants to perform real-time control of a system of interest. In such case, typically, both the state variables $\mathbf{x}$ and the parameter $\boldsymbol{\beta}$ of the model of the system to be controlled need to be estimated.

In the following section, we propose a framework that combines our model structure identification algorithm (Algorithm 1) with classical filtering algorithms to offer a solution to the above mentioned challenges.

IV. Model Refinement

Our method allows inference of model structures that can be decomposed as linear combinations of nonlinear functions chosen from a dictionary set. We note, however, that the identification of parameters nonlinearly embedded in these functions is a non-trivial task. As we saw in the previous section, a naive approach consists in augmenting the set of dictionary functions with various candidate nonlinearities for which nonlinearly embedded parameters are given specific values. We would then rely on the approximation of the true nonlinearities as a linear combination of these dictionary functions, i.e. on estimating the true nonlinearity as an "interpolation" from discretely valued candidate nonlinearities.

On the other hand, filtering methods have been widely used to estimate parameters for a given (nonlinear) parametric structure [7]. The main issue with filtering methods is that they require *a priori* knowledge of such model structures

and cannot easily be used to infer model structures in other ways than by trying individual structures and comparing them using model selection criteria. Typically, this process has a very high computational cost.

In the following section, we show how our model structure inference (described in Algorithm 1) can be used to identify nonlinear terms that can benefit from further refinement using filtering approaches. For example, if the right hand side of one of the equations in the identified model was to contain a linear combination of the form $\frac{0.2684}{1+x} + \frac{0.7292}{1+x^3}$, a new parametric structure could be created where this term is replaced by $\frac{V_{\max}}{K+x^h}$. This new parametric model structure can then serve as the starting point for filtering methods, which are then used to estimate the values of the parameters in the new parametric structure. Furthermore, the parameters identified using our model structure inference method (see Algorithm 1) can be used as initial guesses or priors for the filtering methods.

A. State extension and filtering

As mentioned above, our model structure inference method can be used to identify nonlinear terms that can benefit from further refinement in their structure. Let the new parametric structure obtained through such refinement be given by:

$$\begin{cases} \mathbf{x}_{t+1} = \mathbf{g}(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\gamma}) + \boldsymbol{\upsilon}_t, \\ \mathbf{z}_t = \mathbf{x}_t + \boldsymbol{\eta}_t, \\ \mathbf{x}_1 = g_0, \end{cases} \tag{14}$$

where g_0 is the initial guess of the state vector $\mathbf{x}$.

Extended or unscented Kalman filtering are celebrated methods used to identify both the state variables in $\mathbf{x}$ and the parameters in $\boldsymbol{\gamma}$ of a given parametric model structure such as the one provided in (14). Simultaneous identification of state variables and parameters can be done using a "state extension" approach where constant parameters such as those contained in the model parameter vector $\boldsymbol{\gamma}$ are considered as additional state variables with a rate of change equal to zero. In this way, constant parameters are treated as constant functions of time as opposed to constant numbers [13].

The parameters identified using our model structure inference method (see Algorithm 1) can be used as initial guesses or priors for the parameters $\boldsymbol{\gamma}$.

Filtering can be made more tractable by considering that the unknown parameters $\boldsymbol{\gamma}$ evolve according to a Brownian motion. For this, we introduce a new variable $\boldsymbol{\phi}_k$ and consider the following linear process model:

$$\begin{bmatrix} \boldsymbol{\phi}_{t+1} \\ \boldsymbol{\gamma}_{t+1} \end{bmatrix} = \begin{bmatrix} I & 0 \\ \Delta\tau & I \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_t \\ \boldsymbol{\gamma}_t \end{bmatrix} + \boldsymbol{\varrho}_t \ , \tag{15}$$

where $\boldsymbol{\varrho}_t$ has covariance:

$$\mathbf{Q}_{\boldsymbol{\varrho}} := \sigma^2 \begin{bmatrix} \Delta\tau & \Delta\tau^2/2 \\ \Delta\tau^2/2 & \Delta\tau^3/3 \end{bmatrix} \ ,$$

where σ^2 must be chosen *a priori*. We further define the augmented state parametric structure as:

$$\begin{aligned} \bar{\mathbf{x}}_t &\triangleq \begin{bmatrix} \mathbf{x}_t \\ \boldsymbol{\phi}_t \\ \boldsymbol{\gamma}_t \end{bmatrix}, \qquad \bar{\mathbf{g}}(\bar{\mathbf{x}}_t, \mathbf{u}_t) \triangleq \begin{bmatrix} \mathbf{g}(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\gamma}_t) \\ \boldsymbol{\phi}_t \\ \boldsymbol{\gamma}_t + \Delta t \boldsymbol{\phi}_t \end{bmatrix}, \\ \bar{\boldsymbol{\upsilon}}_t &\triangleq \begin{bmatrix} \boldsymbol{\upsilon}_t \\ \boldsymbol{\varrho}_t \end{bmatrix}, \qquad \bar{g}_0 \triangleq \begin{bmatrix} \mathbf{g}_0 \\ \boldsymbol{\phi}_0 \\ \boldsymbol{\gamma}_0 \end{bmatrix}, \end{aligned} \tag{16}$$

where $\bar{g}_0$ is the initial state estimate.

We can now write the full augmented dynamic model as

$$\begin{cases} \bar{\mathbf{x}}_{t+1} = \bar{\mathbf{g}}(\bar{\mathbf{x}}_t, \mathbf{u}_t) + \bar{\boldsymbol{\upsilon}}_t, \\ \mathbf{z}_t = [\mathbf{I}_{n_{\mathbf{x}}}, \mathbf{0}]\bar{\mathbf{x}}_t + \boldsymbol{\eta}_t, \\ \bar{\mathbf{x}}_1 = \bar{g}_0. \end{cases} \tag{17}$$

The new process noise $\bar{\boldsymbol{\upsilon}}_t$ has positive definite covariance matrix

$$\bar{\mathbf{Q}}_t = \begin{bmatrix} \mathbf{Q}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\boldsymbol{\varrho}} \end{bmatrix} .$$

Using such a state extension approach, the problem of parameter estimation is converted into a problem of state estimation, for which the goal is to estimate the extended state $\bar{\mathbf{x}}$ from measurements of the output $\mathbf{z}$. More precisely, we are trying to determine the initial conditions $\bar{g}_0$, which, when used to initialise the system (14), generates the observed output $\mathbf{z}$.

B. Algorithm combining model structure identification and model refinement

In this section we present Algorithm 2, which constitutes the main algorithm combining 1) our model structure identification method (Algorithm 1) with 2) model refinement of model structures using filtering. Model structure identification is done off-line and thus requires *batched data* (historical sensor measurements) which were collected *a priori*. Once a 'rough' model structure is obtained, model refinement can be performed on-line by feeding *streaming data* (sensor measurements that arrive in real-time), e.g. obtained from the microfludic device in Fig. 1.

In Algorithm 2, we define a trial as the application of the model structure identification procedure described in Algorithm 1 using a given set of dictionary functions and a given regularisation parameter λ.

V. Numerical Simulations

We use the example in our previous work [11] with the same parameters and initial condition settings for both model and Algorithm 1. An eight species generalised repressilator [14] is considered, where each of the species represses another species in a ring topology. The corresponding dynamic equations that we would like to identify from time series data are as follows:

$$\begin{aligned} \frac{dx_{1,t}}{dt} &= \frac{p_{11}}{K_1 + x_{8,t}^{p_{13}}} + p_{14} - p_{15} x_{1,t}, \\ \frac{dx_{i,t}}{dt} &= \frac{p_{i1}}{K_i + x_{i-1,t}^{p_{i3}}} + p_{i4} - p_{i5} x_{i,t}, \ \forall i = 2, \ldots 8, \end{aligned} \tag{18}$$

Algorithm 2 Online Model Selection Algorithm

IDENTIFICATION:
Require: Batched Data
procedure IDENTIFICATION(S trials)
  for $s = 1, \ldots, S$ **do**
    Choose a regularisation parameter λ_s;
    Choose a set of dictionary functions;
    Using the set of dictionary functions, construct $\mathbf{A}_s$ from batched data;
    $\mathcal{M}_s$ = IDENTIFICATION($\lambda_s, \mathbf{A}_s$); % Apply Algorithm 1 to get a model $\mathcal{M}_s$;
  end for
  Pick the top $\hat{S}$ ranked $\mathcal{M}_s$ models based on a certain model selection criterion.
end procedure
Update:
procedure UPDATE($\hat{S}$ trials)
  Update candidate functions as stated in the introduction to Section IV;
end procedure
FILTERING:
Require: Streaming Data
procedure FILTERING($\hat{S}$ trials)
  while New data $\mathbf{z}_t$ is available **do**
    for $s = 1, \ldots, \hat{S}$ **do**
      $\mathcal{M}_s^{\text{new}}$ = FILTERING($\mathbf{z}_t$); % Apply Filtering techniques to refine model $\mathcal{M}_s$
    end for
  end while
  if Not convergent **then goto** *IDENTIFICATION*
  end if
end procedure

where $\frac{dx_{i,t}}{dt}$ is a numerical estimation of the time derivative of $x_{i,t}$ (see appendix of [11] for details).

We assume the mean value for these parameters across different species and experiments are $\bar{p}_{i1} = 40$, $K_i = 1$, $\bar{p}_{i3} = 3$, $\bar{p}_{i4} = 0.5$, $\bar{p}_{i5} = 1$, $\forall i$. We simulate the ODEs in (18) to generate the time series data. In each "experiment" or simulation of (18), the initial conditions are randomly drawn from a standard uniform distribution on the open interval $(0, 1)$. As an example, we have considered that in each experiment parameters of the true system (18) can vary by up to 20% of their mean values and so are drawn from a uniform distribution over $[0.8\bar{p}_{ij}, 1.2\bar{p}_{ij}]$.

The numerical simulation procedure can be summarised as follows:

1) The deterministic system of ODEs (18) is solved numerically with an adaptive fourth-order Runge-Kutta method;
2) Gaussian measurement noise with variance σ^2 is added to the corresponding time-series data obtained in the previous step[1];
3) The data is re-sampled using uniform intervals[2];
4) A dictionary matrix is constructed as illustrated in Section III;
5) Algorithm 1 is used to identify the model.

Following the procedure described in Section III, the candidate dictionary matrix $\mathbf{A}$ in step 5) above is constructed by selecting as candidate nonlinear dictionary functions those typically used to represent terms appearing in ODE models of Gene Regulatory Networks. As a proof of concept, we only consider linear, constant and Hill functions as potential candidate functions. The set of Hill functions with Hill coefficient h, both in activating and repressing form, for the i^{th} state variables at discrete time point t, are:

$$\text{hill}(x_{i,t}, K_i, h_{\text{num}}, h_{\text{den}}) \triangleq \frac{x_{i,t}^{h_{\text{num}}}}{K_i + x_{i,t}^{h_{\text{den}}}} \tag{19}$$

where h_{num} and h_{den} represent the Hill coefficients. When $h_{\text{num}} = 0$, the Hill function has a repression form, whereas an activation form is obtained for $h_{\text{num}} = h_{\text{den}} \neq 0$.

We are interested in identifying the regulation type (linear or Hill type, repression or activation) and the corresponding parameters p_{i1}, the basal expression rate p_{i4} and the degradation rate constant p_{i5}, as well as K_i, $\forall i$. Since there are 8 state variables, we can construct the dictionary matrix $\mathbf{A}$ with 8 (basis functions for linear terms) $+(8 * 8)$ (8 Hill functions with $K_i \in \{0.5, 1.5\}$ and $h_{num}, h_{den} \in \{2, 3\}$, both repression and activation form) $+1$ (constant unit vector) $= 73$ columns. The corresponding matrix $\mathbf{A}$ is given in Eq. (20). Note that none of the Hill functions in the set of dictionary functions has a value of K_i equal to 1.

To quantify the identification accuracy of the algorithm, we use the root of normalised mean square error (RNMSE) as a performance index, i.e. $\textbf{RNMSE} = \|\bar{\boldsymbol{\beta}}_{\text{estimate}} - \bar{\boldsymbol{\beta}}_{\text{true}}\|_2 / \|\bar{\boldsymbol{\beta}}_{\text{true}}\|_2$ where $\bar{\boldsymbol{\beta}}_{\text{true}}$ (resp. $\bar{\boldsymbol{\beta}}_{\text{estimate}}$) represents the average of the C parameters values (resp. the average of the C identified parameters values). Similarly to what we showed in Fig. 1 of [11], we observe that a larger number of experiments C or a larger length of single time series data M leads to a smaller RNMSE value.[3] In our simulation, we take $C = 10$ and $M = 100$. The corresponding RNMSE for the application of Algorithm 1 to the identification of model (18) is RNMSE = 0.047 when 50 independent experiments are considered.

Since the identification procedure for each state variable is independent, we only focus on the identification of the dynamics $\dot{x}_1$. Similar results are obtained for the identification of the other equations Both the linear term x_1 and the constant term can be identified with an average parameter estimation value of $\bar{p}_{14} = 0.501 \approx 0.5$ and $\bar{p}_{15} = 1.07 \approx 1$.

In our result, both dictionary funtions $\frac{1}{0.5+x_{8,t}^3}$ and $\frac{1}{1.5+x_{8,t}^3}$ are selected by Algorithm 1 to be part of the dynamics of $dx_{1,t}/dt$, and the average of the corresponding parameters over $C = 10$ experiments are 8 and 35.7, respectively. This means that $\frac{40}{1+x_{8,t}^3}$ can be approximated by $\frac{8}{0.5+x_{8,t}^3} +$

[1]In the example presented here, for simplicity of exposition, we consider the noiseless case corresponding to $\sigma = 0$.

[2]In this example, the interval length is set to 1.

[3]The RNMSE values for varying values of C and M are not shown here due to space limitation.

$$\mathbf{A} = \begin{bmatrix} x_{11} & \dots & x_{81} & \text{hill}(x_{11}, 0.5, 0, 2) & \dots & \text{hill}(x_{81}, 1.5, 0, 3) & \text{hill}(x_{81}, 1.5, 3, 3) & 1 \\ \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{1M} & \dots & x_{8M} & \text{hill}(x_{1M}, 0.5, 0, 2) & \dots & \text{hill}(x_{8M}, 1.5, 0, 3) & \text{hill}(x_{8M}, 1.5, 3, 3) & 1 \end{bmatrix} \in \mathbb{R}^{M \times 73}. \quad (20)$$

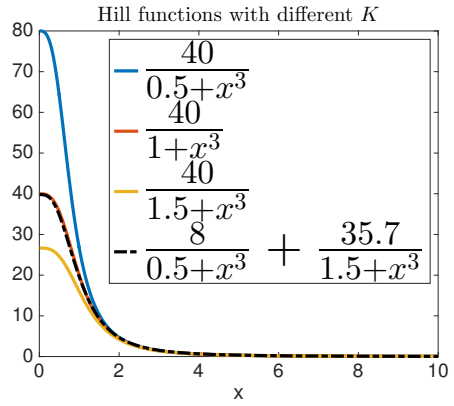


Fig. 3. After refinement iterations (see Figure 4) the parameters γ_1 and γ_2 of the new structure $\frac{\gamma_1}{\gamma_2+x_8^3}$ selected to replace $\frac{8}{0.5+x_8^3}+\frac{35.7}{1.5+x_8^3}$ (identified from Algorithm 1) are estimated to be $\gamma_1 = 39.99$ and $\gamma_2 = 0.9998$, respecti

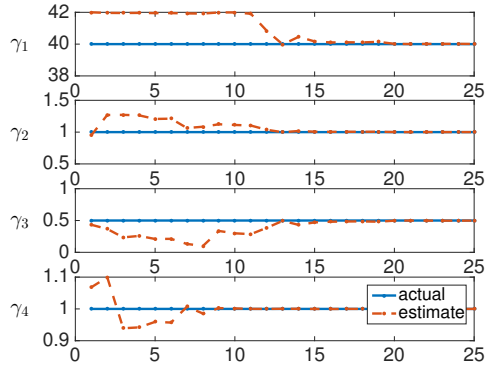


Fig. 4. Evolution of the estimated values of the parameters in equation (21) as a function of the number of streaming data iterations of the Unscented Kalman Filter.

$\frac{35.7}{1.5+x_{8,t}^3}$. The corresponding fitting result can be found in Figure 3.

Next we turn to model refinement ("Filter" in Algorithm 2) using an Unscented Kalman filter [13]. For this, we consider the following equation for dx_1/dt:

$$\frac{dx_{1,t}}{dt} = \frac{\gamma_1}{\gamma_2 + x_{8,t}^3} + \gamma_3 - \gamma_4 x_{1,t}. \quad (21)$$

where $\frac{dx_{1,t}}{dt}$ is a numerical estimation of the time derivative of $x_{1,t}$. where the new parametric structure $\frac{\gamma_1}{\gamma_2+x_{8,t}^3}$ has been used to replace the term $\frac{8}{0.5+x_{8,t}^3}+\frac{35.7}{1.5+x_{8,t}^3}$ that was identified by Algorithm 1. Fig. 4 shows the evolution of the estimated values of the parameters in equation (21) as a function of the number of streaming data iterations of the Unscented Kalman Filter.

VI. Discussion

In this work we presented a novel approach to the identification of the structure and parameters of synthetic gene networks; after introducing the general theoretical framework we applied this approach to the identification of a synthetic oscillator (generalised 8-gene repressilator) with promising results. Ongoing work is focusing on two main directions to extend this work. First, we are investigating the minimal sampling rate necessary to yield adequate numerical estimates of the first derivative dx/dt. Second, further results, not shown in this paper, indicate that RNMSE is high when dynamic noise and measurement noise are high: we are currently working on finer characterisation of the "quality" of the identification in terms of the Signal-to-Noise ratio. Finally, in this paper, we only considered Unscented Kalman Filter. The performance of other filtering techniques will be studied as part of future work.

References

[1] A. Dobrin, P. Saxena, and M. Fussenegger, "Synthetic biology: applying biological circuits beyond novel therapies," *Integrative Biology*, 2016.

[2] L. Ljung, *System Identification: Theory for the User*. Prentice Hall, 1999.

[3] F. Menolascina, G. Fiore, E. Orabona, L. De Stefano, M. Ferry, J. Hasty, M. di Bernardo, and D. di Bernardo, "In-Vivo Real-Time Control of Protein Expression from Endogenous and Synthetic Gene Networks," *PLoS Computational Biology*, vol. 10, no. 5, 2014.

[4] J. Uhlendorf, S. Bottani, F. Fages, P. Hersen, and G. Batt, "Towards real-time control of gene expression: Controlling the hog signaling cascade." in *Pacific Symposium on Biocomputing*. World Scientific, 2011, pp. 338–349.

[5] H.-M. Kaltenbach, S. Dimopoulos, and J. Stelling, "Systems analysis of cellular networks under uncertainty," *FEBS letters*, vol. 583, no. 24, pp. 3923–3930, 2009.

[6] J. Vanlier, C. Tiemann, P. Hilbers, and N. van Riel, "Parameter uncertainty in biochemical models described by ordinary differential equations," *Mathematical biosciences*, vol. 246, no. 2, pp. 305–314, 2013.

[7] E. Walter and L. Pronzato, "Identification of parametric models," *Communications and Control Engineering*, vol. 8, 1997.

[8] J. Ruess, F. Parise, A. Milias-Argeitis, M. Khammash, and J. Lygeros, "Iterative experiment design guides the characterization of a light-inducible gene expression circuit," *Proceedings of the National Academy of Sciences*, vol. 112, no. 26, p. 201423947, 2015.

[9] M. S. Ferry, I. A. Razinkov, and J. Hasty, "Microfluidics for synthetic biology: from design to execution," *Methods in Enzymology*, vol. 497, pp. 295–372, 2011.

[10] Y. Setty, A. Mayo, M. Surette, and U. Alon, "Detailed map of a cis-regulatory input function," *Proceedings of the National Academy of Sciences*, vol. 100, no. 13, p. 7702, 2003.

[11] W. Pan, Y. Yuan, L. Ljung, J. Gonçalves, and G.-B. Stan, "Nonlinear Biochemical Reaction Networks Identification From Heterogeneous Datasetss," in *IEEE 54st Annual Conference on Decision and Control (CDC)*. IEEE, 2015.

[12] W. Pan, Y. Yuan, J. Gonçalves, and G.-B. Stan, "A Sparse Bayesian Approach to the Identification of Nonlinear State-Space Systems," *IEEE Transaction on Automatic Control*, 2015.

[13] R. Van Der Merwe and E. A. Wan, "The square-root unscented kalman filter for state and parameter-estimation," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, vol. 6. IEEE, 2001, pp. 3461–3464.

[14] N. Strelkowa and M. Barahona, "Switchable genetic oscillator operating in quasi-stable mode," *Journal of The Royal Society Interface*, p. rsif20090487, 2010.