Identification of Nonlinear State-Space Systems from Heterogeneous Datasets

Wei Pan, Ye Yuan, Lennart Ljung, Jorge Gonçalves and Guy-Bart Stan

***Abstract*—This paper proposes a new method to identify nonlinear state-space systems from heterogeneous datasets. The method is described in the context of identifying biochemical/gene networks (i.e., identifying both reaction dynamics and kinetic parameters) from experimental data. Simultaneous integration of various datasets has the potential to yield better performance for system identification. Data collected experimentally typically vary depending on the specific experimental setup and conditions. Typically, heterogeneous data are obtained experimentally through (a) replicate measurements from the same biological system or (b) application of different experimental conditions such as changes/perturbations in biological inductions, temperature, gene knock-out, gene over-expression, etc. We formulate here the identification problem using a Bayesian learning framework that makes use of "sparse group" priors to allow inference of the sparsest model that can explain the whole set of observed, heterogeneous data. To enable scale up to large number of features, the resulting non-convex optimisation problem is relaxed to a re-weighted Group Lasso problem using a convex-concave procedure. As an illustrative example of the effectiveness of our method, we use it to identify a genetic oscillator (generalised eight species repressilator). Through this example we show that our algorithm outperforms Group Lasso when the number of experiments is increased, even when each single time-series dataset is short. We additionally assess the robustness of our algorithm against noise by varying the intensity of process noise and measurement noise.**

I. Introduction

The problem of identifying biological networks from experimental time-series data is of fundamental interest in systems and synthetic biology [1]–[3]. Tools from system identification [4] can be applied for such purposes. However, most system identification methods produce estimates of model parameters based on data coming from a single experiment.

The interest in identification methods able to handle several datasets simultaneously is twofold. Firstly, with the increasing availability of "big data" obtained from sophisticated biological instruments, e.g., large 'omics' datasets, attention has turned to the efficient and effective integration of these data and to the maximum extraction of information from them. Such datasets typically contain (a) data from replicates of an experiment performed on a biological system of interest under identical experimental conditions, or (b) data measured from a biochemical network subjected to different experimental conditions, for example, different biological inducers, temperature, stress factors, gene knock-out or gene over-expression. The challenges for simultaneously considering heterogeneous datasets during system identification are: (a) the system itself is unknown, i.e., neither the structure nor the corresponding parameters are known; (b) it is unclear how heterogeneous datasets collected under different experimental conditions influence the "quality" of the identified system; (c) each single time-series data may be short. These second and third points are particularly important as biological experiments become increasingly costly in time and resources when long time-series dataset are required. Furthermore, repeat or perturbation experiments may be conducted over different time ranges, with different sampling frequencies, under various conditions, and in different laboratories, which likely affects the success of identification.

Another important consideration comes from the purpose of dynamic models. Highly detailed or complex models are typically difficult to handle using rigorous control design methods. 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 model class uncertainties and parameter uncertainties [5], [6]. To assess these uncertainties, replicates of multiple experiments are typically necessary.

In the context of biology, the use of kinetic models to understand the function of biological systems has already been successfully illustrated in [7], [8]. Furthermore, the use of heterogeneous dataset during system identification has been proposed as a means to improve the accuracy of genetic regulatory network reconstruction methods [9]. Typically, biological experiments are accompanied by a set of corresponding reference control experiments, whose profiles are used to determine differential gene expression [10], [11]. Modern techniques try to harness the "wisdom of crowd" concept by integrating the predictions from multiple datasets into a single reconstructed network termed the "consensus gene regulatory network" [12], [13]. For instance, in [13], the authors grouped the algorithms by applying the Euclidean distance on the confidence scores of the links in the inferred networks. They showed that integration

Dr Wei Pan gratefully acknowledges the support of Microsoft Research through the PhD Scholarship Program for his stay at Imperial College London. Dr Guy-Bart Stan gratefully acknowledges the support of the EPSRC grant EP/P009352/1 and of the EPSRC Fellowship for Growth EP/M002187/1. (Corresponding author: Prof. Ye Yuan)

W. Pan is with the Department of Bioengineering, Imperial College London, United Kingdom and with DJI Innovations, Shenzhen, China. Email: w.pan11@imperial.ac.uk.

Y. Yuan is with School of Automation, Huazhong University of Science and Technology, China. Email: yye@hust.edu.cn.

L. Ljung is with Division of Automatic Control, Department of Electrical Engineering, Linköping University, Sweden. Email: ljung@isy.liu.se.

J. Gonçalves is with the Control Group, Department of Engineering, University of Cambridge, United Kingdom and with the Luxembourg Centre for Systems Biomedicine, Luxembourg. Email: jmg77@cam.ac.uk.

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

The corresponding code is available at https://github.com/panweihit/BSID.

of diverse algorithms outperformed each individual inference methods. The consensus network was obtained in three ways: average of the estimated coefficients over conditions, *a priori* biological knowledge, and pre-calculated coefficients obtained from the application of a Gaussian graphical model [14] on the combined data sets. However, the problem of accurate reconstruction of gene regulatory networks is far from fully resolved. Recent works [15], [16] advanced the state-of-art by using new type of regularisation techniques. Unfortunately, the dynamical models considered so far have been mostly constrained to linear systems, an assumption that is rarely satisfied by biological systems.

Our approach is based on the concept of sparse Bayesian learning [2], [17], [18] and on the definition of a unified optimisation problem allowing model identification from heterogeneous datasets, and whose solution is a model consistent with all datasets available for identification. The ability to consider various datasets simultaneously can potentially avoid non-identifiability issues arising when a single dataset is used [19].

The main contributions of this paper are as follows:

- Formulation of a nonlinear identification problem using datasets from heterogeneous experiments.
- Derivation of a sparse Bayesian formulation of this identification problem by introducing "sparse group" priors.
- Relaxation of the resulting non-convex optimisation problem using a convex-concave procedure and development of an efficient iterative reweighted Group Lasso algorithm that allows to solve large problems defined through a large number of features.

The paper is organised as follows. In section II-A, we introduce the nonlinear model class considered in the paper. In section II-C, the identification problem from heterogeneous datasets is formulated. In section III, the identification problem from heterogeneous datasets is cast as a non-convex Bayesian learning problem with structured sparse priors. A convex reweighted Group Lasso type algorithm is then derived as a relaxation of its non-convex counterpart. In section IV, results from identification of a generalised eight species repressilator system are provided as an example application of our method. In the end, we conclude the paper and give directions for further work in section V.

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. $\|\mathbf{w}\|_1$ and $\|\mathbf{w}\|_2$ denote the ℓ_1 and ℓ_2 norm of the vector $\mathbf{w}$, respectively. $\|\mathbf{w}\|_0$ denotes the ℓ_0 norm of the vector $\mathbf{w}$, which counts the number of nonzero elements in the vector $\mathbf{w}$. $\mathrm{diag}\,[\gamma_1, \ldots, \gamma_N]$ denotes a diagonal matrix with principal diagonal elements being $\gamma_1, \ldots, \gamma_N$. $\mathbb{E}(\boldsymbol{\alpha})$ stands for the expectation of the stochastic variable $\boldsymbol{\alpha}$. $\propto$ means "proportional to". $\mathrm{blkdiag}[\mathbf{A}^{[1]}, \ldots, \mathbf{A}^{[C]}]$ denotes a block diagonal matrix with principal diagonal blocks being $\mathbf{A}^{[1]}, \ldots, \mathbf{A}^{[C]}$ in turn. $\mathrm{Tr}(\mathbf{A})$ 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 non-negative.

II. Problem Formulation

A. Model

We consider dynamical systems described by nonlinear differential/difference equation with additive process noise:

$$\begin{aligned}\delta(x_{nt}) &= \mathbf{f}_n(\mathbf{x}_t, \mathbf{u}_t)\mathbf{v}_n + \xi_{nt} \quad n = 1, \ldots, n_{\mathbf{x}} \\ &= \sum\nolimits_{s=1}^{N_n} v_{ns} f_{ns}(\mathbf{x}_t, \mathbf{u}_t) + \xi_{nt},\end{aligned} \tag{1}$$

where $\mathbf{x}_t$ is the state variable, $\mathbf{u}_t$ is the external control input; x_{nt} represent the n-th state variable at time t (similar for u_{nt}); $\delta(x_{nt}) = \dot{x}_{nt}$ for continuous-time system; $\delta(x_{nt}) = x_{nt}$ or $x_{nt} - x_{n,t-1}$ or some *known* transformation of historical data for discrete-time system; $v_{ns} \in \mathbb{R}$ and $f_{ns}(\mathbf{x}_t, \mathbf{u}_t) : \mathbb{R}^{n_{\mathbf{x}}+n_{\mathbf{u}}} \to \mathbb{R}$ and $\mathbf{v}_n$ are *basis functions* and corresponding parameters respectively that govern the dynamics, where $n_{\mathbf{x}}$ and $n_{\mathbf{u}}$ are the dimension of $\mathbf{x}$ and $\mathbf{u}$ respectively. The functions $f_{ns}(\mathbf{x}_t, \mathbf{u}_t)$ are assumed to be Lipschitz continuous. ξ_{nt} represents additive process noise, which is assumed to be i.i.d. Gaussian. Note that we do not assume *a priori* knowledge of the form of the nonlinear functions appearing on the right-hand side of the equations in (1), e.g., whether the degradation obeys first-order or enzymatic catalysed dynamics or whether the proteins are repressors or activators.

B. Heterogeneous Time-Series Datasets

In what follows, we will assume data are sampled from a total number M of time instances, and that the state variables and their first derivatives/differences are recorded into a set $\mathcal{D}$:

$$\mathcal{D} = \{x_{nt}, \delta(x_{nt})\}_{n=1,\ldots,n_{\mathbf{x}};t=1,\ldots,M}. \tag{2}$$

By inputting the dataset $\mathcal{D}$ into eq. (1), we get the following regression problem formulation the for n-th state variable

$$\mathbf{y}_n = \boldsymbol{\Psi}_n \mathbf{v}_n + \boldsymbol{\xi}_n, \ n = 1, \ldots, n_{\mathbf{x}}, \tag{3}$$

where

$$\begin{aligned}\mathbf{y}_n &\triangleq [\delta(x_{n1}), \ldots, \delta(x_{nM})]^\top \in \mathbb{R}^{M\times 1} \\ \mathbf{v}_n &\triangleq [v_{n1}, \ldots, v_{nN_n}]^\top \in \mathbb{R}^{N_n\times 1} \\ \boldsymbol{\xi}_n &\triangleq [\xi_{n1}, \ldots, \xi_{nM})]^\top \in \mathbb{R}^{M\times 1}.\end{aligned}$$

$\boldsymbol{\Psi}_n \in \mathbb{R}^{M\times N_n}$ is defined as a *dictionary matrix* whose j-th column is $[f_{nj}(\mathbf{x}_1, \mathbf{u}_1), \ldots, f_{nj}(\mathbf{x}_M, \mathbf{u}_M)]^\top$. The process noise or disturbance vector $\boldsymbol{\xi}_n$ is assumed to be Gaussian distributed with zero mean and covariance $\boldsymbol{\Pi} \in \mathbb{R}_+^{M\times M}$[1]. The identification goal is to estimate $\mathbf{v}_n$ in the linear regression problem formulated in eq. (3).

If a total number of C (named after the first letter of word "collection") datasets are collected from C independent experiments, we put a subscript $[c]$ to indicate the identification problem associated with the specific dataset obtained from

[1]Note that the covariance matrix is not necessarily diagonal.

experiment $[c]$, with $c \in \{1, \dots, C\}$. Similar to $\mathcal{D}$ in (2), we define the dataset for experiment $[c]$ as

$$\begin{aligned}\mathcal{D}^{[c]} &= \{x_{nt}^{[c]}, \delta^{[c]}(x_{nt})\}_{c=1,\dots,C;n=1,\dots,n_{\mathbf{x}};t=1,\dots,M^{[C]}} \\ &= \{\mathbf{x}_t^{[c]}, \delta^{[c]}(\mathbf{x}_t)\}_{c=1,\dots,C;t=1,\dots,M^{[C]}}.\end{aligned} \tag{4}$$

In what follows we gather in a matrix $\mathbf{A}_n^{[c]}$ similar to $\mathbf{\Psi}_n$ the set of *all* candidate dictionary functions that we want to consider during the identification. The identification problem is then written as:

$$\mathbf{y}_n^{[c]} = \mathbf{A}_n^{[c]}\mathbf{w}_n^{[c]} + \boldsymbol{\xi}_n^{[c]}, \quad n = 1, \dots, n_{\mathbf{x}}, \ c = 1, \dots, C. \tag{5}$$

Since the $n_{\mathbf{x}}$ linear regression problems in (5) are independent, for simplicity of notation, we omit the subscript n used to index the state variable and simply write:

$$\mathbf{y}^{[c]} = \mathbf{A}^{[c]}\mathbf{w}^{[c]} + \boldsymbol{\xi}^{[c]}, \ c = 1, \dots, C, \tag{6}$$

in which

$$\begin{aligned}\mathbf{A}^{[c]} &\triangleq \left[\mathbf{A}_{:,1}^{[c]}, \dots, \mathbf{A}_{:,N}^{[c]}\right] \\ &= \begin{bmatrix} f_1\left(\mathbf{x}_1^{[c]}, \mathbf{u}_1^{[c]}\right) & \dots & f_N\left(\mathbf{x}_1^{[c]}, \mathbf{u}_1^{[c]}\right) \\ \vdots & & \vdots \\ f_1\left(\mathbf{x}_{M^{[c]}}^{[c]}, \mathbf{u}_{M^{[c]}}^{[c]}\right) & \dots & f_N\left(\mathbf{x}_{M^{[c]}}^{[c]}, \mathbf{u}_{M^{[c]}}^{[c]}\right) \end{bmatrix} \\ &\quad \in \mathbb{R}^{M^{[c]} \times N}, \\ \mathbf{w}^{[c]} &\triangleq \left[w_1^{[c]}, \dots, w_N^{[c]}\right]^\top \in \mathbb{R}^N, \\ \boldsymbol{\xi}^{[c]} &\triangleq \left[\xi_1^{[c]}, \dots, \xi_{M^{[c]}}^{[c]})\right]^\top \in \mathbb{R}^{M^{[c]}},\end{aligned} \tag{7}$$

where $\mathbf{x}_t^{[c]} = \left[x_{1t}^{[c]}, \dots, x_{n_{\mathbf{x}}t}^{[c]}\right] \in \mathbb{R}^{n_{\mathbf{x}}}$ is the state vector at time instant t. It should be noted that N, the number of dictionary functions or number of columns of the dictionary matrix $\mathbf{A}^{[c]} \in \mathbb{R}^{M^{[c]} \times N}$, can be very large. Without loss of generality, we assume $M^{[1]} = \dots = M^{[C]} = M$.

C. Identification Setup for Heterogeneous Datasets

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 is obtained from different experimental conditions, it is usually very hard to combine the resulting heterogeneous 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.

To consider heterogeneous datasets in one single formulation, we stack in eq. (8) the various individual equations in (6).

In eq. (8), $\mathbf{A}_i = \text{blkdiag}[\mathbf{A}_{:,i}^{[1]}, \dots, \mathbf{A}_{:,i}^{[C]}]$, and $\mathbf{w}_i = [w_i^{[1]}, \dots, w_i^{[C]}]^\top$, for $i = 1, \dots, N$. Based on the stacked formulation given in eq. (8) we further define

$$\begin{aligned}\mathbf{y} &= \begin{bmatrix} \mathbf{y}^{[1]} \\ \vdots \\ \mathbf{y}^{[C]} \end{bmatrix}, \mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \mid \cdots \mid \mathbf{A}_N \end{bmatrix}, \\ \mathbf{w} &= \left[\begin{array}{c} \mathbf{w}_1 \\ \hline \vdots \\ \hline \mathbf{w}_N \end{array}\right], \boldsymbol{\xi} = \left[\begin{array}{c} \boldsymbol{\xi}^{[1]} \\ \hline \vdots \\ \hline \boldsymbol{\xi}^{[C]} \end{array}\right],\end{aligned} \tag{9}$$

which gives

$$\mathbf{y} = \mathbf{A}\mathbf{w} + \boldsymbol{\xi}. \tag{10}$$

This yields a formulation that is very similar to the previous linear regression problem for a single dataset in eq. (3). However, there is a key difference: there is a special block structure for $\mathbf{y}$, $\mathbf{A}$ and $\mathbf{w}$ in the multi-experiment formulation (10).

Remark 1: When $\mathbf{w}^{[i]}$ is fixed to be $\mathbf{w}$ for all the experiments, i.e., $\mathbf{w}^{[1]} = \dots = \mathbf{w}^{[C]} = \mathbf{w}$, we can formulate the identification problem as a single linear regression problem by concatenation:

$$\begin{bmatrix} \mathbf{y}^{[1]} \\ \vdots \\ \mathbf{y}^{[C]} \end{bmatrix} = \begin{bmatrix} \mathbf{A}^{[1]} \\ \vdots \\ \mathbf{A}^{[C]} \end{bmatrix} \mathbf{w} + \begin{bmatrix} \boldsymbol{\xi}^{[1]} \\ \vdots \\ \boldsymbol{\xi}^{[C]} \end{bmatrix}. \tag{11}$$

III. Methods

To get an estimate of $\mathbf{w}$ in (10), we use Bayesian modeling to treat all unknowns as stochastic variables with certain probability distributions [29]. For $\mathbf{y} = \mathbf{A}\mathbf{w} + \boldsymbol{\xi}$, it is assumed that the stochastic variables in the vector $\boldsymbol{\xi}$ are Gaussian distributed with *unknown* covariance matrix $\mathbf{\Pi}$, i.e., $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Pi})$.

In what follows we consider the following variable substitution for the inverse of the unknown covariance matrix: $\mathbf{S} \triangleq \mathbf{\Pi}^{-1}$. In such a case, using the properties of Gaussian distributions, the likelihood of the output $\mathbf{y}$ given the parameter $\mathbf{w}$ is

$$\mathcal{P}(\mathbf{y}|\mathbf{w}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{w}, \mathbf{\Pi}) \tag{12}$$

$$\propto \exp\left[-\frac{1}{2}(\mathbf{A}\mathbf{w} - \mathbf{y})^\top \mathbf{S}(\mathbf{A}\mathbf{w} - \mathbf{y})\right]. \tag{13}$$

A. Sparsity Inducing Priors

In Bayesian models, a prior distribution $\mathcal{P}(\mathbf{w})$ can be defined as

$$\mathcal{P}(\mathbf{w}) = \prod_{i=1}^{N} \mathcal{P}(\mathbf{w}_i)$$

$$
\begin{aligned}
\begin{bmatrix} \mathbf{y}^{[1]} \\ \vdots \\ \mathbf{y}^{[C]} \end{bmatrix} &= \underbrace{\left[\begin{array}{ccc|c|ccc} \mathbf{A}_{:,1}^{[1]} & \dots & \mathbf{A}_{:,N}^{[1]} & & & & \\ & & & \ddots & & & \\ & & & & \mathbf{A}_{:,1}^{[C]} & \dots & \mathbf{A}_{:,N}^{[C]} \end{array}\right]}_{C \text{ Blocks}} \left[\begin{array}{c} \mathbf{w}^{[1]} \\ \hline \vdots \\ \hline \mathbf{w}^{[C]} \end{array}\right] + \left[\begin{array}{c} \boldsymbol{\xi}^{[1]} \\ \hline \vdots \\ \hline \boldsymbol{\xi}^{[C]} \end{array}\right] \\
&= \underbrace{\left[\begin{array}{ccc|c|ccc} \mathbf{A}_{:,1}^{[1]} & & & & \mathbf{A}_{:,N}^{[1]} & & \\ & \ddots & & \ddots & & \ddots & \\ & & \mathbf{A}_{:,1}^{[C]} & & & & \mathbf{A}_{:,N}^{[C]} \end{array}\right]}_{N \text{ Blocks}} \left[\begin{array}{c} w_1^{[1]} \\ \vdots \\ \hline w_1^{[C]} \\ \hline \vdots \\ \hline w_N^{[1]} \\ \vdots \\ w_N^{[C]} \end{array}\right] + \left[\begin{array}{c} \boldsymbol{\xi}^{[1]} \\ \hline \vdots \\ \hline \boldsymbol{\xi}^{[C]} \end{array}\right] \\
&= \left[\begin{array}{c|c|c} \mathbf{A}_1 & \cdots & \mathbf{A}_N \end{array}\right] \left[\begin{array}{c} \mathbf{w}_1 \\ \hline \vdots \\ \hline \mathbf{w}_N \end{array}\right] + \left[\begin{array}{c} \boldsymbol{\xi}^{[1]} \\ \hline \vdots \\ \hline \boldsymbol{\xi}^{[C]} \end{array}\right].
\end{aligned} \tag{8}
$$

where

$$
\begin{aligned}
\mathcal{P}(\mathbf{w}_i) &\propto \exp\left[-\frac{1}{2}\sum_{j=1}^{C} g(w_i^{[j]})\right] \\
&= \prod_{j=1}^{C} \exp\left[-\frac{1}{2} g(w_i^{[j]})\right] \\
&= \prod_{j=1}^{C} \mathcal{P}(w_i^{[j]}),
\end{aligned}
$$

with $g(w_i^{[j]})$ being a given function of $w_i^{[j]}$. Generally, $\mathbf{w}$ in (10) is sparse, and therefore certain sparsity properties should be enforced on $\mathbf{w}$. To this effect, the function $g(\cdot)$ is usually chosen to be a concave, non-decreasing function of $\left|w_i^{[j]}\right|$ [18]. Examples of such functions $g(\cdot)$ include Generalised Gaussian priors and Student's *t* priors (see [18], [30] for details).

Computing the posterior mean $\mathbb{E}(\mathbf{w}|\mathbf{y})$ is typically intractable because the posterior $\mathcal{P}(\mathbf{w}|\mathbf{y})$ is highly coupled and non-Gaussian. To alleviate this problem, ideally one would like to approximate $\mathcal{P}(\mathbf{w}|\mathbf{y})$ as a Gaussian distribution for which efficient algorithms to compute the posterior exist [29]. For this, the introduction of lower bounding *super-Gaussian* priors $\mathcal{P}(w_i^{[j]})$, i.e., $\mathcal{P}(w_i^{[j]}) = \max_{\gamma_i>0} \mathcal{N}(w_i^{[j]}|0,\gamma_i)\varphi(\gamma_i)$, can be used to obtain an analytical approximation of $\mathcal{P}(\mathbf{w}|\mathbf{y})$ [30].

Note that current problem (10) has a block structure as pointed out in the previous section, i.e., the solution $\mathbf{w}$ is expected to be block-wise sparse. Therefore, sparsity promoting priors should be specified for $\mathcal{P}(\mathbf{w}_i)$, $\forall i$. To do this, for each block $\mathbf{w}_i$, we define a hyper-parameter γ_i such that

$$
\mathcal{P}(\mathbf{w}_i) = \max_{\gamma_i>0} \mathcal{N}(\mathbf{w}_i|\mathbf{0},\gamma_i\mathbf{I}_C)\varphi(\gamma_i) \tag{14}
$$

$$
= \max_{\gamma_i>0} \prod_{j=1}^{C} \mathcal{N}(w_i^{[j]}|0,\gamma_i)\varphi(\gamma_i), \tag{15}
$$

where $\varphi(\gamma_i)$ is a nonnegative function, which is treated as a hyperprior with γ_i being its associated hyperparameter. Throughout, we call $\varphi(\gamma_i)$ the "*potential function*". This Gaussian relaxation is possible if and only if $\log \mathcal{P}(\sqrt{w_i})$ is concave on $(0,\infty)$. Defining

$$
\begin{aligned}
\boldsymbol{\gamma}_i &= [\gamma_i,\dots,\gamma_i] \in \mathbb{R}^C,\ \boldsymbol{\Gamma}_i = \operatorname{diag}[\boldsymbol{\gamma}_i], \\
\boldsymbol{\gamma} &= [\boldsymbol{\gamma}_1,\dots,\boldsymbol{\gamma}_N] \in \mathbb{R}^{NC},\ \boldsymbol{\Gamma} = \operatorname{diag}[\boldsymbol{\gamma}],
\end{aligned} \tag{16}
$$

we have

$$
\mathcal{P}(\mathbf{w}) = \prod_{i=1}^{N} \mathcal{P}(\mathbf{w}_i) = \max_{\boldsymbol{\gamma}>\mathbf{0}} \mathcal{N}(\mathbf{w}|\mathbf{0},\boldsymbol{\Gamma})\varphi(\boldsymbol{\gamma}). \tag{17}
$$

B. Cost Function

Using the Gaussian likelihood introduced in eq. (13) and the variational prior in eq. (17), we can define the following optimisation problem jointly on $\mathbf{w}$, $\boldsymbol{\gamma}$ and $\mathbf{S}$.

Proposition 1: The unknowns $\mathbf{w},\boldsymbol{\gamma},\mathbf{S}$ can be obtained by solving the following optimisation problem

$$
\begin{aligned}
&\mathcal{L}(\mathbf{w},\boldsymbol{\gamma},\mathbf{S}) \\
&= \min_{\mathbf{w},\boldsymbol{\gamma},\mathbf{S}} \{-\log|\mathbf{S}| + \log|\boldsymbol{\Gamma}| + \log|\boldsymbol{\Gamma}^{-1} + \mathbf{A}^\top\mathbf{S}\mathbf{A}| \\
&\quad + (\mathbf{y}-\mathbf{A}\mathbf{w})^\top \mathbf{S}(\mathbf{y}-\mathbf{A}\mathbf{w}) + \mathbf{w}^\top\boldsymbol{\Gamma}^{-1}\mathbf{w} + \sum_{j=1}^{N} p(\gamma_j)\},
\end{aligned} \tag{18}
$$

where $\boldsymbol{\Gamma}$ is given in eq. (16).

Proof: To derive the cost function in eq. (18), we first introduce the posterior mean and covariance:

$$
\mathbf{m}_\mathbf{w} = \boldsymbol{\Sigma}_\mathbf{w}\mathbf{A}^\top\mathbf{S}\mathbf{y}, \tag{19}
$$

$$
\boldsymbol{\Sigma}_\mathbf{w} = (\mathbf{A}^\top\mathbf{S}\mathbf{A} + \boldsymbol{\Gamma}^{-1})^{-1}. \tag{20}
$$

Since the data likelihood $\mathcal{P}(\mathbf{y}|\mathbf{w})$ is Gaussian,

$$
\begin{aligned}
&\mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{w},\mathbf{S}^{-1}) \\
&= \frac{1}{(2\pi)^{M/2}|\mathbf{S}|^{-1/2}} \exp\left[-\frac{1}{2}(\mathbf{y}-\mathbf{A}\mathbf{w})^\top\mathbf{S}(\mathbf{y}-\mathbf{A}\mathbf{w})\right],
\end{aligned} \tag{21}
$$

and we can write the marginal likelihood as

$$\begin{aligned}&\int \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{w},\mathbf{\Pi})\mathcal{N}(\mathbf{w}|\mathbf{0},\mathbf{\Gamma})\prod_{j=1}^{N}\varphi(\gamma_j)d\mathbf{w}\\ =&\frac{1}{(2\pi)^{M/2}\,|\mathbf{S}|^{-1/2}}\frac{1}{(2\pi)^{N}}\int \exp\{-E(\mathbf{w})\}d\mathbf{w}\prod_{j=1}^{N}\varphi(\gamma_j),\end{aligned} \tag{22}$$

where

$$E(\mathbf{w}) = \frac{1}{2}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right) + \frac{1}{2}\mathbf{w}^{\top}\mathbf{\Gamma}^{-1}\mathbf{w}. \tag{23}$$

Equivalently, we get

$$E(\mathbf{w}) = \frac{1}{2}(\mathbf{w}-\mathbf{m}_{\mathbf{w}})^{\top}\mathbf{\Sigma}_{\mathbf{w}}^{-1}(\mathbf{w}-\mathbf{m}_{\mathbf{w}}) + E(\mathbf{y}), \tag{24}$$

where $\mathbf{m}_{\mathbf{w}}$ and $\mathbf{\Sigma}_{\mathbf{w}}$ are given by eq. (19) and (20).

We first show the data-dependent term $E(\mathbf{y})$ is jointly convex in $\mathbf{w}$ and $\boldsymbol{\gamma}$. From eq. (19) and (20), the data-dependent term can be re-expressed as

$$\begin{aligned}E(\mathbf{y}) =&\frac{1}{2}\left(\mathbf{y}^{\top}\mathbf{S}\mathbf{y} - \mathbf{y}^{\top}\mathbf{S}\mathbf{A}\mathbf{\Sigma}_{\mathbf{w}}\mathbf{A}^{\top}\mathbf{S}\mathbf{y}\right)\\ =&\frac{1}{2}\left(\mathbf{y}^{\top}\mathbf{S}\mathbf{y} - \mathbf{y}^{\top}\mathbf{S}\mathbf{A}\mathbf{\Sigma}_{\mathbf{w}}\mathbf{\Sigma}_{\mathbf{w}}^{-1}\mathbf{\Sigma}_{\mathbf{w}}\mathbf{A}^{\top}\mathbf{S}\mathbf{y}\right)\\ =&\frac{1}{2}\left(\mathbf{y}-\mathbf{A}\mathbf{m}_{\mathbf{w}}\right)^{\top}\mathbf{S}\left(\mathbf{y}-\mathbf{A}\mathbf{m}_{\mathbf{w}}\right) + \frac{1}{2}\mathbf{m}_{\mathbf{w}}^{\top}\mathbf{\Gamma}^{-1}\mathbf{m}_{\mathbf{w}}\\ =&\min_{\mathbf{w}}\left[\frac{1}{2}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right) + \frac{1}{2}\mathbf{w}^{\top}\mathbf{\Gamma}^{-1}\mathbf{w}\right].\end{aligned} \tag{25}$$

Using (24), we can evaluate the integral in (22) and get

$$\int \exp\{-E(\mathbf{w})\}d\mathbf{w} = \exp\{-E(\mathbf{y})\}(2\pi)^{N}|\mathbf{\Sigma}_{\mathbf{w}}|^{1/2}. \tag{26}$$

Applying a $-2\log(\cdot)$ transformation to eq. (22), we obtain eq. (27). Therefore we get the cost function in eq. (18) to be minimised over $\mathbf{w}, \boldsymbol{\gamma}, \mathbf{S}$. ∎

C. Algorithm

It is easy to check that the cost function in eq. (18) is convex in $\mathbf{w}$ and $\mathbf{S}$ but concave in $\mathbf{\Gamma}$. This non-convex optimisation problem can be formulated as a convex-concave procedure (CCCP). It can be shown that solving this CCCP is equivalent to solving a series of iterative convex optimisation programs, which is guaranteed to converge to a stationary point [31]. Let

$$\begin{aligned}u(\mathbf{w},\boldsymbol{\gamma},\mathbf{S}) \triangleq&\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)\\ &+\mathbf{w}^{\top}\mathbf{\Gamma}^{-1}\mathbf{w} - \log\det\mathbf{S},\\ v(\boldsymbol{\gamma},\mathbf{S}) \triangleq&-\left[\log|\mathbf{\Gamma}| + \log|\mathbf{\Gamma}^{-1} + \mathbf{A}^{\top}\mathbf{S}\mathbf{A}| + \sum_{j=1}^{N}p(\gamma_j)\right].\end{aligned} \tag{28}$$

It is easy to check that $v(\boldsymbol{\gamma},\mathbf{S})$ is a convex function with respect to $\boldsymbol{\gamma}$. Furthermore, $\log|\cdot|$ is concave in the space of positive semi-definite matrices. Since we adopt a super-Gaussian prior with potential function $\varphi(\gamma_j), \forall j$, as described in (15), a direct consequence is that $p(\gamma_j) = -\log\varphi(\gamma_j)$ is concave, and, therefore, $-p(\gamma_j)$ is convex [17] (if the prior is chosen as a Student's t prior, then $p(\gamma_j) = 1$). Note that $u(\mathbf{w},\boldsymbol{\gamma},\mathbf{S})$ is jointly convex in $\mathbf{w}$, $\boldsymbol{\gamma}$ and $\mathbf{S}$, while $v(\boldsymbol{\gamma},\mathbf{S})$ is jointly convex in $\boldsymbol{\gamma}$ and $\mathbf{S}$. As a consequence, the minimisation of the objective function can be formulated as a concave-convex procedure:

$$\min_{\boldsymbol{\gamma}\succeq\mathbf{0},\mathbf{S}\succeq\mathbf{0},\mathbf{w}} u(\mathbf{w},\boldsymbol{\gamma},\mathbf{S}) - v(\boldsymbol{\gamma},\mathbf{S}). \tag{29}$$

Since $v(\boldsymbol{\gamma},\mathbf{S})$ is differentiable over $\boldsymbol{\gamma}$, the problem in eq. (29) can be transformed into the following iterative convex optimisation problem

$$\mathbf{w}^{k+1} = \operatorname*{argmin}_{\mathbf{w}} u(\mathbf{w},\boldsymbol{\gamma}^{k},\mathbf{S}^{k}) \tag{30}$$

$$\boldsymbol{\gamma}^{k+1} = \operatorname*{argmin}_{\boldsymbol{\gamma}\succeq\mathbf{0}} u(\mathbf{w}^{k},\boldsymbol{\gamma},\mathbf{S}^{k}) - \nabla_{\boldsymbol{\gamma}}v(\boldsymbol{\gamma}^{k},\mathbf{S}^{k})^{\top}\boldsymbol{\gamma} \tag{31}$$

$$\mathbf{S}^{k+1} = \operatorname*{argmin}_{\mathbf{S}\succeq\mathbf{0}} u(\mathbf{w}^{k},\boldsymbol{\gamma}^{k},\mathbf{S}) - \nabla_{\mathbf{S}}v(\boldsymbol{\gamma}^{k},\mathbf{S}^{k})^{\top}\mathbf{S}. \tag{32}$$

Using basic principles in convex analysis, we then obtain the following analytic form for the negative gradient of $v(\boldsymbol{\gamma})$ at $\boldsymbol{\gamma}$ (using the chain rule):

$$\begin{aligned}\boldsymbol{\alpha}^{k} \triangleq&-\nabla_{\boldsymbol{\gamma}}v(\boldsymbol{\gamma},\mathbf{S}^{k})^{\top}|_{\boldsymbol{\gamma}=\boldsymbol{\gamma}^{k}}\\ =&\nabla_{\boldsymbol{\gamma}}\left[\log|\mathbf{\Gamma}^{-1} + \mathbf{A}^{\top}\mathbf{S}^{k}\mathbf{A}| + \log|\mathbf{\Gamma}|\right]\\ =&\operatorname{diag}\{\left[(\mathbf{\Gamma}^{k})^{-1} + \mathbf{A}^{\top}\mathbf{S}^{k}\mathbf{A}\right]^{-1}\}\cdot\operatorname{diag}\{-(\mathbf{\Gamma}^{k})^{-2}\}\\ &+\operatorname{diag}^{-1}\{\mathbf{\Gamma}^{k}\}\\ =&\underbrace{\left[\begin{array}{c|c|c}\boldsymbol{\alpha}_{11}^{k} & \cdots & \boldsymbol{\alpha}_{1N}^{k}\end{array}\right]}_{N\ \textbf{Blocks}}\\ =&\left[\begin{array}{c|c|c}\underbrace{\alpha_{11}^{k},\ldots,\alpha_{11}^{k}}_{C\ \textbf{Elements}} & \cdots & \underbrace{\alpha_{1N}^{k},\ldots,\alpha_{1N}^{k}}_{C\ \textbf{Elements}}\end{array}\right].\end{aligned} \tag{33}$$

Therefore, the iterative procedures in eq. (30) and (31) for $\mathbf{w}^{k+1}$ and $\boldsymbol{\gamma}^{k+1}$, respectively, can be formulated as

$$\begin{aligned}\left[\mathbf{w}^{k+1},\boldsymbol{\gamma}^{k+1}\right] = \operatorname*{argmin}_{\boldsymbol{\gamma}\succeq\mathbf{0},\mathbf{w}}&\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}^{k}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)\\ &+\sum_{i=1}^{N}\left(\frac{\mathbf{w}_i^{\top}\mathbf{w}_i}{\gamma_i} + C\gamma_i\alpha_i^{k}\right).\end{aligned} \tag{34}$$

The optimal $\boldsymbol{\gamma}$ components can be computed analytically as $\gamma_i = \frac{\|\mathbf{w}_i\|_2}{\sqrt{C\alpha_i^{k}}}$. Once $\boldsymbol{\gamma}$ is fixed, we can compute $\mathbf{w}^{k+1}$ by solving the following optimisation problem:

$$\min_{\mathbf{w}}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}^{k}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right) + 2\sum_{i=1}^{N}\|\theta_i^{k}\cdot\mathbf{w}_i\|_2, \tag{35}$$

where $\theta_i^{k} = C\alpha_i^{k}$. We can then inject this into the expression of γ_i, which yields

$$\gamma_i^{k+1} = \frac{\|\mathbf{w}_i^{k+1}\|_2}{\sqrt{C\alpha_i^{k}}}. \tag{36}$$

After we get $\mathbf{w}^{k+1}$ and $\boldsymbol{\gamma}^{k+1}$, we can proceed with the optimisation iteration in (32):

$$\begin{aligned}\Lambda^{k} =&-\nabla_{\mathbf{S}}v(\boldsymbol{\gamma}^{k},\mathbf{S}^{k})\\ =&\nabla_{\mathbf{S}}\left(\log\det\left(\mathbf{\Gamma}^{-k} + \mathbf{A}^{\top}\mathbf{S}^{k}\mathbf{A}\right)\right)\\ =&\mathbf{A}(\mathbf{\Gamma}^{-k} + \mathbf{A}^{\top}\mathbf{S}^{k}\mathbf{A})^{-1}\mathbf{A}^{\top}.\end{aligned} \tag{37}$$

$$
\begin{aligned}
&-2\log\left[\frac{1}{(2\pi)^{M/2}\left|\mathbf{S}\right|^{-1/2}}\frac{1}{(2\pi)^{N}}\int\exp\{-E(\mathbf{w})\}d\mathbf{w}\prod_{j=1}^{N}\varphi(\gamma_j)\right] \\
&=(M+2N)\log 2\pi-\log\left|\mathbf{S}\right|+\log\left|\mathbf{\Gamma}\right|+\log\left|\mathbf{\Gamma}^{-1}+\mathbf{A}^{\top}\mathbf{S}\mathbf{A}\right|+\sum_{j=1}^{N}p(\gamma_j)+\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)+\mathbf{w}^{\top}\mathbf{\Gamma}^{-1}\mathbf{w}.
\end{aligned}
\tag{27}
$$

Letting $\mathbf{Y}^{k+1}=(\mathbf{A}\mathbf{w}^{k+1}-\mathbf{y})\cdot(\mathbf{A}\mathbf{w}^{k+1}-\mathbf{y})^{\top}$, we can get an estimate of the inverse of covariance matrix $\mathbf{S}$ as:

$$
\mathbf{S}^{k+1}=\underset{\mathbf{S}\succeq\mathbf{0}}{\operatorname{argmin}}\operatorname{Tr}\left(\mathbf{S}\mathbf{Y}^{k+1}\right)-\log\det\mathbf{S}+\operatorname{Tr}\left(\Lambda^{k}\mathbf{S}\right). \tag{38}
$$

Given $\boldsymbol{\gamma}^{k+1}$ in (36) and $\mathbf{S}^{k+1}$ in (38), we can then go back to (33) to update $\boldsymbol{\alpha}$ for the next iteration.

The iterative identification procedure described above is summarised in Algorithm 1.

Algorithm 1 Nonlinear Identification Algorithm using Heterogeneous Datasets

1: Collect C heterogeneous groups of time-series data from the system of interest (assuming the system can be described by (1));
2: Select the candidate dictionary functions that will be used to construct the dictionary matrix described in Section II-C;
3: Initialise $\theta_i^0=1,\ \forall i,\ \alpha_i^0=\frac{\theta_i^0}{C}$, $\mathbf{S}^0=\mathbf{I}$, $\Lambda^0=\mathbf{I}$;
4: **for** $k=0,\ldots,k_{\max}$ **do**
5: $\mathbf{w}^{k+1}$ can be obtained by solving the following weighted minimisation problem over $\mathbf{w}$

$$
\min_{\mathbf{w}}\frac{1}{2}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}^{k}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)+\sum_{i=1}^{N}\|\theta_i^{k}\cdot\mathbf{w}_i\|_2; \tag{39}
$$

6: Update γ_i^{k+1} using eq. (36);
7: Let $\mathbf{Y}^{k+1}=(\mathbf{A}\mathbf{w}^{k+1}-\mathbf{y})\cdot(\mathbf{A}\mathbf{w}^{k+1}-\mathbf{y})^{\top}$;
8: $\mathbf{S}^{k+1}$ can be obtained by solving the following weighted minimisation problem over the inverse of the covariace matrix:

$$
\min_{\mathbf{S}\succeq\mathbf{0}}\operatorname{Tr}\left(\mathbf{Y}^{k+1}+\Lambda^{k}\right)\mathbf{S}-\log\det\mathbf{S}; \tag{40}
$$

9: Update $\boldsymbol{\alpha}^{k+1}$ using eq. (33);
10: Update $\theta_i^{k+1}=C\alpha_i^{k+1}$;
11: Update Λ^{k+1} using eq. (37);
12: **if** a stopping criterion is satisfied **then**
13: Break;
14: **end if**
15: **end for**

Remark 2: 1) It should be noted that when noise is Gaussian i.i.d. with *known* variance, sparse Bayesian learning algorithms are provably better than classic Group Lasso algorithms in terms of mean square error [32].

2) The initialisation step is important (line 3 of in Algorithm 1). In special cases where the process noise in (1) is Gaussian i.i.d and there is no measurement noise, $\mathbf{S}$ can be fixed to $\lambda^{-1}\mathbf{I}$ for all k, where λ is a positive real number, i.e., no update through eq. (40) is carried out. In such situations, λ can be treated as the equivalent of the regularisation/trade-off parameter in the Group Lasso algorithm described by eq. (40) and cross validation can be implemented through variations of the initialisation values.
3) When the model obtained is used for prediction purposes, the inverse covariance estimation procedure in eq. (40) can be used to quantify the prediction uncertainty or "risk".
4) Essentially, Algorithm (1) consists of a reweighted Group Lasso algorithm (39) and a reweighted inverse covariance estimation algorithm (40). Both problems are convex and can be implemented using many numerical optimisation algorithms [37] such as the Alternating Direction Method of Multipliers (ADMM) [33], [38].

D. Connection to Semidefinite Programming Formulations and the Sparse Multiple Kernel Method

The iteration in eq. (34) can be rewritten in the following compact form

$$
\begin{aligned}
\left[\mathbf{w}^{k+1},\boldsymbol{\gamma}^{k+1}\right]=\underset{\boldsymbol{\gamma}\succeq\mathbf{0},\mathbf{w}}{\operatorname{argmin}}&\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right)^{\top}\mathbf{S}^{k}\left(\mathbf{y}-\mathbf{A}\mathbf{w}\right) \\
&+\mathbf{w}^{\top}\mathbf{\Gamma}^{-1}\mathbf{w}-\nabla_{\boldsymbol{\gamma}}v(\boldsymbol{\gamma}^{k},\mathbf{S}^{k})^{\top}\boldsymbol{\gamma}.
\end{aligned}
\tag{41}
$$

Using the standard procedure in [34], this is equivalent to the following Semidefinite Programming optimisation problem:

$$
\begin{aligned}
\min_{\mathbf{z},\mathbf{w},\boldsymbol{\gamma}}\quad & \mathbf{z}-\nabla_{\boldsymbol{\gamma}}v(\boldsymbol{\gamma}^{k},\mathbf{S}^{k})^{\top}\boldsymbol{\gamma} \\
\text{subject to}\quad & \begin{bmatrix}\mathbf{z} & (\mathbf{y}-\mathbf{A}\mathbf{w})^{\top} & \mathbf{w}^{\top}\\ \mathbf{y}-\mathbf{A}\mathbf{w} & (\mathbf{S}^{k})^{-1} & \mathbf{0}\\ \mathbf{w} & \mathbf{0} & \mathbf{\Gamma}\end{bmatrix}\succeq\mathbf{0} \\
& \boldsymbol{\gamma}\succeq\mathbf{0}
\end{aligned}
$$

Solving this Semidefinite Programming optimisation is too costly for all but problems with a small number of variables. This means that the number of samples and the dimension of the system cannot be too large simultaneously. In this Semidefinite Programming formulation, $\mathbf{\Gamma}$ is closely related to the sparse multiple kernel presented in [35]. Certain choices of kernels may introduce some good properties or help reduce algorithmic complexity. In our case, we choose $\mathbf{\Gamma}$ to have a diagonal or a DC kernel structure.

IV. Simulations

In this section, we use numerical simulations to show the effectiveness of the proposed algorithm. To compare the identification accuracy of the algorithms considered for comparison, we use the root of normalised mean square error (RNMSE) as a performance index, i.e.,

$$\textbf{RNMSE} = \frac{\|\mathbf{w}_{\text{estimate}} - \mathbf{w}_{\text{true}}\|_2}{\|\mathbf{w}_{\text{true}}\|_2}.$$

Several factors affect the RNMSE, e.g., number of experiments C, measurement noise intensity, dynamic noise intensity, length of single time-series data M, and number of candidate dictionary functions N. In what follows, we shall focus on showing results pertaining to RNMSE when the number of experiment, C, and the length of single time-series, M, for each experiment are varied.

As an illustrative example, we consider a model of an eight species generalised repressilator [36], which is a system where each of the species represses another species in a ring topology. The corresponding dynamic equations are as follows:

$$\begin{aligned}\dot{x}_{1t} &= \frac{p_{11}}{p_{12}^{p_{13}} + x_{8t}^{p_{13}}} + p_{14} - p_{15}x_{1t},\\ \dot{x}_{it} &= \frac{p_{i1}}{p_{i2}^{p_{i3}} + x_{i-1,t}^{p_{i3}}} + p_{i4} - p_{i5}x_{it},\ \forall i = 2,\dots 8,\end{aligned} \tag{42}$$

where $p_{ij},\ i = 1,\dots,8,\ j = 1,\dots,5$. We assume the mean value for these parameters across different species and experiments are $\bar{p}_{i1} = 40$, $\bar{p}_{i2} = 1$, $\bar{p}_{i3} = 3$, $\bar{p}_{i4} = 0.5$, $\bar{p}_{i5} = 1$, $\forall i$. We simulate the ODEs in (42) to generate the time-series data. In each "experiment" or simulation of (42), the initial conditions are randomly drawn from a standard uniform distribution on the open interval $(0, 1)$. The parameters in each experiment vary no more than 20% from the mean values.[2]

Following the procedure described in the previous sections, candidate nonlinear dictionary functions need to be considered from the set of nonlinear functions typically used in ODE models of Gene Regulatory Networks. As an illustrative example, we will hereafter only consider Hill functions as potential nonlinear candidate functions. The set of Hill functions with Hill coefficient h, both in activating and repressing form, for the i^{th} state variables at time instant t are:

$$\text{hill}(x_{it}, K, h_{\text{num}}, h_{\text{den}}) \triangleq \frac{x_{it}^{h_{\text{num}}}}{K^{h_{\text{den}}} + x_{it}^{h_{\text{den}}}} \tag{43}$$

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 $h_{\text{num}} = h_{\text{den}} \neq 0$.

In our identification experiment, we assume h_{num}, h_{den} and K to be known. We are interested in identifying the regulation type (linear or Hill type, repression or activation) and its corresponding multiplying parameter p_{i1}, the basal expression rate p_{i4}, and the degradation rate constant p_{i5}, $\forall i$. Since there are 8 state variables, we can construct the dictionary matrix $\mathbf{A}$ with 8 (dictionary functions for linear terms) $+(2 * 8)$ (dictionary functions for Hill functions, both repression and activation form) $+1$ (constant unit vector) $= 25$ columns. The corresponding matrix $\mathbf{A} \in \mathbb{R}^{M\times 25}$ is given in eq. (45).

Typically in a state-space representation, a measurement equation is often considered to take into account the measurement dynamics and how noise enters it. For the purpose of our simulations, we have consider the simple case of additive measurement noise on the state variables:

$$z_{nt} = x_{nt} + \epsilon_{nt} \tag{44}$$

where ϵ_{nt} is the measurement noise assumed to be i.i.d. Gaussian with zero mean and bounded standard deviation.

The numerical simulation procedure that we used to produce time-series data for the purpose of identification can be summarised as follows:

1) The deterministic system of ODEs (42) is discretised using an Euler method with sampling time 0.1;
2) We consider two scenarios: the first noiseless, while the second noisy, for which we vary the standard deviation of process noise (i.e., ξ_{nt} in eq. (1)) and measurement noise (i.e., ϵ_{nt} in eq. (44));
3) A dictionary matrix is constructed as explained above;
4) We run both Group Lasso and Algorithm 1 with the maximal iteration number defined as $k_{\max} = 5$ (see step 4 in Algorithm 1) to identify the model then compare the identification performance in terms of RNMSE.

For the noiseless case, we varied the number of experiments C and length of single time-series M. For a fixed C and M, we computed the RNMSE over 50 simulations by varying initial conditions and parameters p_{ij}. The RNMSE for various values of C and M are shown in Fig. 1(a) and Fig. 1(b). Expectedly, identification using single short time-series data is challenging for both approaches. However, integration of multiple (even shorter) time-series data using our algorithm offers significantly improved identification performance as can be seen when higher values of C are used. As observed in Fig. 1, the RNMSE decreases when either the number of experiments C or the length of single time-series M increases. When both C and M are high enough, e.g., $C = 10$, $M = 100$, the RNMSE approaches zero.

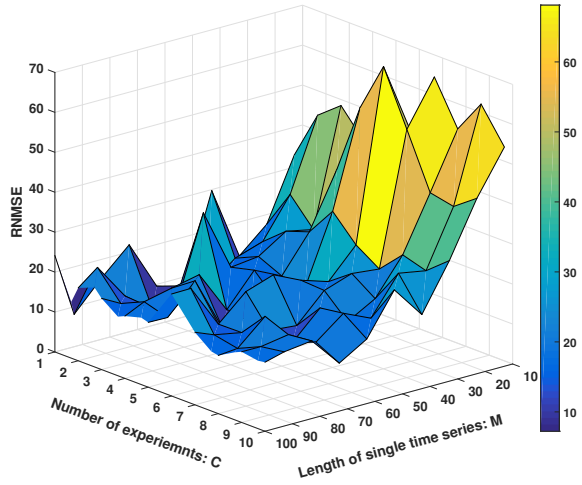
For the noisy case, we varied the standard deviation of process noise, ξ_{nt} in (1), and measurement noise, ϵ_{nt} in (44), while fixing the number of experiments C to 10 and length of single experiment M to 100. For a fixed intensity of process and measurement noises, we compute the RNMSE over 50 simulations by varying initial conditions and parameters p_{ij}. The RNMSE for various noise combinations are shown in Fig. 2(a) and Fig. 2(b). As can be seen, our algorithm clearly outperforms Group Lasso in terms of RNMSE.

V. Conclusion and Discussion

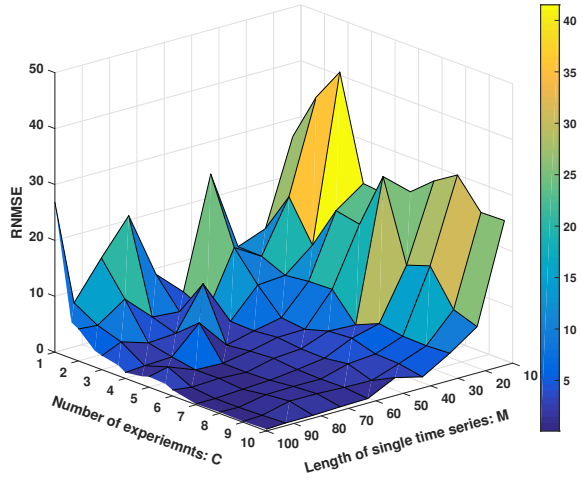
System identification for nonlinear state-space systems is not a trivial task. In our previous work [2], this problem has been considered using single time-series data. Using our method, it was observed that, as the number of considered

[2]In MATLAB, one can use `pbar_ij*(0.8 + 0.4*rand(1))` to generate easily the corresponding parameters for each experiment.

$$\mathbf{A} = \begin{bmatrix} x_{11} & \dots & x_{81} & \text{hill}(x_{11},1,0,3) & \dots & \text{hill}(x_{81},1,3,3) & 1 \\ \vdots & & \vdots & \vdots & & \vdots & \vdots \\ x_{1M} & \dots & x_{8M} & \text{hill}(x_{1M},1,0,3) & \dots & \text{hill}(x_{8M},1,3,3) & 1 \end{bmatrix}. \tag{45}$$

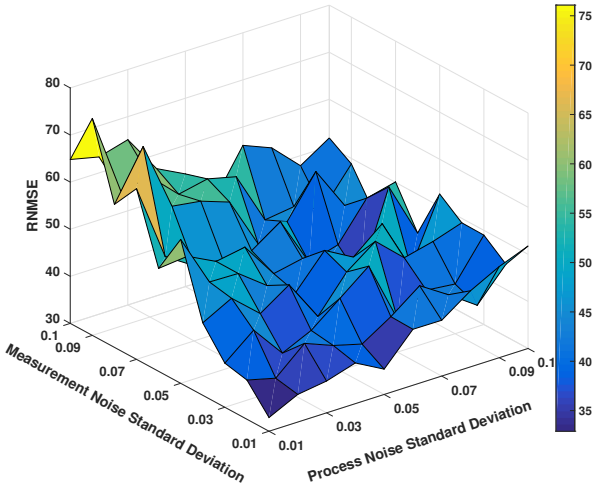


(a) Group Lasso (first iteration of Algorithm 1).

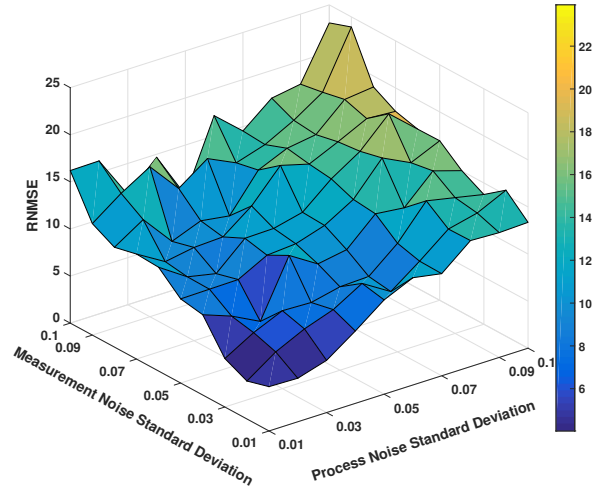


(b) Algorithm 1 with maximal iteration number $k_{\max} = 5$.

Fig. 1. Algorithm comparison in terms of RNMSE averaged over 50 independent experiments by varying number of experiments and length of single experiment.



(a) Group Lasso (first iteration of Algorithm 1).



(b) Algorithm 1 with maximal iteration number $k_{\max} = 5$.

Fig. 2. Algorithm comparison in terms of RNMSE averaged over 50 independent experiments by varying the standard deviation of process noise and measurement noise while fixing the number of experiments to 10 and length of single experiment to 100.

dictionary functions is increased, more data samples are needed for identification. However, even when long time-series data are used, the information contained in the corresponding dataset may not be rich enough for successful identification, e.g., no additional information is provided through longer time series data when steady state is reached. Meanwhile, obtaining long time-series data is typically difficult or costly experimentally as it requires setups allowing to following the time evolution of the system under consideration for long, uninterrupted periods of time.

These issues motivate the simultaneous integration of datasets from various experiments. Data collected experimentally typically vary depending on the specific experimental setup and conditions. Typically, heterogeneous data are obtained experimentally through (a) replicate measurements from the same biological system or (b) application of different experimental conditions such as changes/perturbations in biological inductions, temperature, gene knock-out, gene overexpression, etc. Another important issue is the quantification

of the uncertainty of the identified model for the purpose of further prediction, control or decision making. To tackle these issues, we have formulated here the identification problem using a Bayesian learning framework that makes use of "sparse group" priors to allow inference of the sparsest model that can explain the whole set of observed, heterogeneous data. In the simulated example, our algorithm demonstrably outperforms Group Lasso when the number of experiments is increased, even when each single time-series dataset is short. Additionally, our algorithm is more robust to both process noise and measurement noise compared with Group Lasso methods.

In what follows, we briefly discuss important aspects that need to be considered and further extensions of the method we have presented here. The first important aspect concerns measurement noise. The problem formulation presented in section II-A can be modified to allow consideration of measurement dynamics and associated measurement noise. For the sake of simplicity of exposition, we discuss hereafter the case of a scalar system, i.e., $n = 1$ in (1), where additive i.i.d. Gaussian noise is also considered. In this case, we measurement equation amounts to: $z_t = x_t + \epsilon_t$, where the measurement noise ϵ_t is assumed i.i.d. Gaussian. In the continuous time scalar case, the system in eq. (1) can be equivalently written as $\dot{x}_t = g(x_t) + \xi_t$. We can simply use Taylor series expansion to expand $g(x_t)$ noting that by definition $x_t = y_t - \epsilon_t$:

$$\begin{aligned} g(x_t) &= g(z_t - \epsilon_t) \\ &= g(z_t) \underbrace{- g'(z)|_{z=z_t}\epsilon_t + \mathcal{O}(\epsilon_t^2)}_{\textbf{Correlated Gaussian noise}} \\ &= g(z_t) + \bar{\xi}_t. \end{aligned}$$

If we can estimate $\dot{x}_t$ from y_t properly, e.g., using a Gaussian Process estimation procedure [22], we can then write the following

$$\dot{x}_t^{\textbf{estimate}} = g(z_t) + \bar{\xi}_t + \xi_t = g(z_t) + \eta_t.$$

In this case, the new noise η_t is multiplicative and thus not i.i.d. anymore. Taking inspiration from the Generalised Method of Moment (GMM) [23], [24], we can use the approach described here to determine the form of g, and then estimate the empirical covariance of η_t

The second aspect of importance concerns the selection of the dictionary functions $f_i(.,.)$ in (6). Adequate selection of the dictionary function set is key to the success of the identification. Some prior knowledge of the field for which the models are developed can be helpful here. Indeed, depending on the field for which the dynamical model needs to be built, only a few typical nonlinearities specific to this field need to be considered. For example, the class of models that arise from biochemical reaction networks typically involves nonlinearities that capture fundamental biochemical kinetic laws, e.g., first-order functions $f([S]) = [S]$, mass action functions $f([S_1], [S_2]) = [S_1] \cdot [S_2]$, Michaelis-Menten functions $f([S]) = V_{\max}[S]/(K + [S])$, or Hill functions $f([S]) = V_{\max}[S]^h/(K^h + [S]^h)$.

The third aspect of importance concerns the estimation of the first derivative of the state variables (see Eq. (2) and Eq. (4)), which is not trivial. Estimating time derivatives in continuous-time systems can either be achieved using a measurement equipment with a sufficiently high sampling rate, or using state-of-the-art mathematical approaches [20]. Proper estimation of derivatives is key to the identification procedure [20]. As pointed out in [21], the identification problem is generally solved through discretisation of the proposed model. Assuming that samples are taken at sufficiently short time intervals, various discretisation methods can be applied. Typically, a forward Euler discretisation is used to approximate first order derivatives, i.e., $\mathbf{y}_i$ can be defined as $\mathbf{y}_i \triangleq \left[\frac{x_{i2}-x_{i1}}{\Delta t}, \ldots, \frac{x_{i,M+1}-x_{iM}}{\Delta t}\right]^\top \in \mathbb{R}^{M\times 1}$. In this paper, the local polynomial regression framework in [20] is applied to estimate time derivatives.[3]

VI. Future work

There are several extension of this work that we plan to explore as part of future works.

First, assuming all states and their derivatives in continuous time can be measured or approximated, many identification problems can be formulated as linear regressions. As future work, we plan to extend our framework to partially observable systems and to establish the minimal sampling rate necessary to yield adequate numerical estimates of the first order derivative (see eq. (2) and eq. (4)).

A second aspect we are considering for future work is to better understand the impact of length of observations on identification performance. How can we bound the length of observations to obtain an expected performance? How can we bound the performance given a fixed length of observations? In a linear model, these questions can be answered by analysing the observability of the system. Numerous studies are tackling the question of observability of nonlinear systems (see for example [39]), which can be used to provide theoretical methods and algorithms for determining bounds on the number of observations or the identification performance. Performance guarantees are typically given under the assumption of "non-correlation" or " near-orthogonality" between the columns of the dictionary matrix. This is however hardly satisfied in practice as the dictionary matrix is constructed from data and the variability and randomness in the corresponding dataset cannot guarantee such condition *a priori*.

Finally, we so far only tested our method using simulated data to allow for a fair comparison with the "ground truth" of the system to be identified. In the future, we plan to apply our method to real datasets from biological experiments.

VII. Acknowledgement

The authors would like to thank Dr Aivar Sootla and Dr Tianshi Chen for helpful discussions.

[3]Forward Euler discretisation and central difference discretisation are special cases of the local polynomial regression framework.

REFERENCES

[1] W. Pan, Y. Yuan, J. Gonçalves, and G.-B. Stan, "Reconstruction of arbitrary biochemical reaction networks: A compressive sensing approach," in *IEEE 51st Annual Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 2334–2339.

[2] W. Pan, Y. Yuan, J. Gonçalves, and G.-B. Stan, "A sparse Bayesian approach to the identification of nonlinear state-space systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 1, pp. 182–187, 2016.

[3] Y. Yuan, G. Stan, S. Warnick, and J. Goncalves, "Robust dynamical network structure reconstruction," *Special Issue on System Biology, Automatica*, vol. 47, pp. 1230–1235, 2011.

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

[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] H. De Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *Journal of Computational Biology*, vol. 9, no. 1, pp. 67–103, 2002.

[8] A. F. Villaverde and J. R. Banga, "Reverse engineering and identification in systems biology: strategies, perspectives and challenges," *Journal of the Royal Society Interface*, vol. 11, no. 91, p. 20130505, 2014.

[9] C. Sima, J. Hua, and S. Jung, "Inference of gene regulatory networks using time-series data: a survey," *Current Genomics*, vol. 10, no. 6, pp. 416–429, 2009.

[10] G. K. Smyth *et al.*, "Linear models and empirical bayes methods for assessing differential expression in microarray experiments," *Stat Appl Genet Mol Biol*, vol. 3, no. 1, p. 3, 2004.

[11] F. Rapaport, R. Khanin, Y. Liang, M. Pirun, A. Krek, P. Zumbo, C. E. Mason, N. D. Socci, and D. Betel, "Comprehensive evaluation of differential gene expression analysis methods for rna-seq data," *Genome Biology*, vol. 14, no. 9, p. 3158, 2013.

[12] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, M. Kellis, J. J. Collins, G. Stolovitzky *et al.*, "Wisdom of crowds for robust gene network inference," *Nature Methods*, vol. 9, no. 8, pp. 796–804, 2012.

[13] T. Hase, S. Ghosh, R. Yamanaka, and H. Kitano, "Harnessing diversity towards the reconstructing of large scale gene regulatory networks," *PLoS Computational Biology*, vol. 9, no. 11, p. e1003361, 2013.

[14] J. Schäfer and K. Strimmer, "An empirical Bayes approach to inferring large-scale gene association networks," *Bioinformatics*, vol. 21, no. 6, pp. 754–764, 2005.

[15] N. Omranian, J. M. Eloundou-Mbebi, B. Mueller-Roeber, and Z. Nikoloski, "Gene regulatory network inference using fused lasso on multiple data sets," *Scientific Reports*, vol. 6, 2016.

[16] K. Y. Lam, Z. M. Westrick, C. L. Müller, L. Christiaen, and R. Bonneau, "Fused regression for multi-source gene regulatory network inference," *Plos Computational Biology*, vol. 12, no. 12, p. e1005157, 2016.

[17] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *The Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[18] D. Wipf, B. Rao, and S. Nagarajan, "Latent variable Bayesian models for promoting sparsity," *IEEE Transactions on Information Theory*, vol. 57, no. 9, pp. 6236–6255, 2011.

[19] N. T. Ingolia and J. S. Weissman, "Systems biology: reverse engineering the cell," *Nature*, vol. 454, no. 7208, pp. 1059–1062, 2008.

[20] K. De Brabanter, J. De Brabanter, B. De Moor, and I. Gijbels, "Derivative estimation with local polynomial fitting," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 281–301, 2013.

[21] A. Papachristodoulou and B. Recht, "Determining interconnections in chemical reaction networks," in *American Control Conference*. IEEE, 2007, pp. 4872–4877.

[22] C. E. Rasmussen, "Gaussian processes for machine learning," 2006.

[23] L. P. Hansen, "Large sample properties of generalized method of moments estimators," *Econometrica: Journal of the Econometric Society*, pp. 1029–1054, 1982.

[24] L. P. Hansen and K. J. Singleton, "Generalized instrumental variables estimation of nonlinear rational expectations models," *Econometrica: Journal of the Econometric Society*, pp. 1269–1286, 1982.

[25] S. Boyd, L. El Ghaoul, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. Society for Industrial Mathematics, 1987, vol. 15.

[26] R. Horn and C. Johnson, *Matrix analysis*. Cambridge university press, 1990.

[27] V. Cerone, D. Piga, and D. Regruto, "Enforcing stability constraints in set-membership identification of linear dynamic systems," *Automatica*, vol. 47, no. 11, pp. 2488–2494, 2011.

[28] M. Zavlanos, A. Julius, S. Boyd, and G. Pappas, "Inferring stable genetic networks from steady-state data," *Automatica*, vol. 47, no. 6, pp. 1113–1122, 2011.

[29] C. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, 2006, vol. 4.

[30] J. Palmer, D. Wipf, K. Kreutz-Delgado, and B. Rao, "Variational EM algorithms for non-Gaussian latent variable models," *Advances in Neural Information Processing Systems (NIPS)*, vol. 18, pp. 1059, 2005.

[31] B. K. Sriperumbudur and G. R. Lanckriet, "On the convergence of the concave-convex procedure." in *NIPS*, vol. 9, 2009, pp. 1759–1767.

[32] A. Aravkin, J. V. Burke, A. Chiuso, and G. Pillonetto, "Convex vs non-convex estimators for regression and sparse estimation: the mean squared error properties of ard and glasso," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 217–252, 2014.

[33] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[34] S. Boyd and L. Vandenberghe, *Convex optimisation*. Cambridge university press, 2004.

[35] T. Chen, M. Andersen, L. Ljung, A. Chiuso, and G. Pillonetto, "System identification via sparse multiple kernel-based regularization using sequential convex optimization techniques," *IEEE Transactions on Automatic Control*, vol. 59, no. 11, pp. 2933–2945, 2014.

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

[37] J. Nocedal and S. Wright, *Numerical optimization*. Springer, 2006.

[38] W. Pan, A. Sootla, and G.-B. Stan, "Distributed Reconstruction of Nonlinear Networks: An ADMM Approach," *The International Federation of Automatic Control Cape Town, South Africa*, 2014.

[39] Y. Xue, S. Pequito, J. R. Coelho, P. Bogdan, and G. J. Pappas, "Minimum number of sensors to ensure observability of physiological systems: A case study," in *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2016, pp. 1181–1188.