EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Control Systems Research Group

Learning local modules in dynamic networks without prior topology information

Master Thesis
Master Systems & Control

by
Venkatakrishnan C. Rajagopal

Commissioned by Professor : prof. dr. ir. Paul Van den Hof
Group & Chair : Control Systems
Date of final presentation : 27-October-2020

Internal Supervisor : ir. Karthik R Ramaswamy

Eindhoven

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

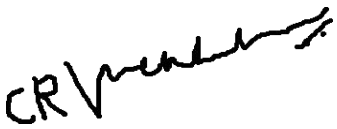
I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date 10/28/20

Name Venkatakrishnan Comandoor Rajagopal

ID-number 1301349

Signature 

Insert this document in your Master Thesis report (2nd page) and submit it on Sharepoint

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU

Learning local modules in dynamic networks without prior topology information

Venkatakrishnan C. Rajagopal, *Student*

***Abstract*— The problem of identifying a single module embedded in a network relies on apriori topology information for selecting the predictor inputs and predicted outputs. In this paper, we address the problem of identifying a single module embedded in a network that is excited through process noise that may be correlated, without any prior information about the nodal or noise topology. Using non-causal Wiener filters, it is shown that all the nodes that contain information about the output node can be detected and the remaining nodes can be dropped to reduce the network size without affecting the single module estimation. A new focused search is conducted for identifying the nodal and noise topology that minimizes a cost function. This search avoids testing all possible combinations and thereby reduces the number of search steps to find the minima. The developed algorithm uses a series of convex optimizations with parallel computation capabilities to obtain the topology without a high computational requirement. Suggestions are made based on the available data for a suitable choice of the cost function. The identified topology is used to build the predictor model using the local direct method to ensure appropriate signal selection under correlated process noise. To avoid model order selection which is computationally complex and to reduce the number of nuisance parameters that affect the target module estimate, the regularized kernel based methods EBDM and EBLDM are used. Numerical simulations illustrate the potential of the developed algorithm.**

***Index Terms*— Dynamic Networks, Gaussian Process, Wiener Filter, Complexity Criteria, Empirical Bayes, Network Reconstruction, Correlated noise**

I. Introduction

A dynamic network is an interconnection of various systems interconnected with each other and can be defined as a set of measurable signals (known as node signals) interconnected through linear time-invariant (LTI) dynamic systems (known as modules), possibly driven by external excitation signals and process noise. Over the past decade, increasing attention has been given to the data-driven modeling of such dynamic networks among the system identification community. The three major problems in the data-driven modeling of dynamic networks are full network identification, topology identification and local module identification.

Full network identification focuses on identifying the whole network dynamics [1]–[8] and the underlying challenge of identifiability [9]–[11]. Topology identification focuses on identifying the nodal interconnection structure of the network [12]–[18]. Local module identification focuses on identifying the dynamics of a single module embedded in a network with a known network and noise topology [19]–[26].

In this study, we focus on identifying the dynamics of a single module embedded in a network with an unknown network and noise topology. Existing literature chooses appropriate predictor inputs and predicted outputs of the predictor model based on the network and noise topology. In [19], the *direct method* for single module identification in dynamic networks is introduced by extending the direct method for closed loop identification [27]. In this method, a Multi-Input Single-Output (MISO) predictor model is built with all node-in-neighbor of the output node as the predictor input signals. Therefore, according to [19] only the set of nodes that are connected to the output node is needed for consistent identification of the target module.

However, the target module can be consistently identified with limited number of predictor input signals using a MISO predictor model. Using a limited number of predictor inputs can introduce confounding variables[1], which cause bias in the target module estimation. This issue can be handled by adding additional predictor inputs. The algorithm for limited predictor input selection is provided in [28] with conditions for handling confounding variables in [22]. The estimates obtained using the above methods are consistent and exhibit *Maximum Likelihood* (ML) properties. However, these methods for generating predictor models are applicable only for dynamic networks that have uncorrelated process noise.

The situation of process noise correlation in a dynamic network introduces confounding variables to the target module estimation that can not be mitigated by extending the predictor input signals in the direct method. Such a problem can be handled by the *Indirect method* [29] and its variants like the *Two Stage method* [19], [28] by projecting the nodal signals on external excitation signals and building a MISO predictor model. However, these indirect methods rely on the presence of external excitation signals and adding these signals increase

This project has received funding from the European Research Council (ERC), Advanced Research Grant SYSDYNET, under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 694504).

The author is with the University of Technology, Eindhoven. {v.comandoor.rajagopal@student.tue.nl}

[1]unmeasured variables that directly or indirectly influence the input and output of an estimation problem

the cost of the experiment. In addition, the indirect methods do not exhibit ML properties.

To handle the effect of confounding variables arising from correlated process noise, the *local direct method* [26] provides a solution. Based on the *noise correlation structure (or noise topology)* and nodal topology of a subset of nodes, the local direct method provides algorithms for appropriately selecting the predictor inputs and predicted outputs of the predictor model. Thereby, the MISO predictor model moves to a Multi-Input Multi-Output (MIMO) predictor model and the estimates of the modules are obtained using the prediction error method (PEM). In order to identify one module in the MIMO setup, all the modules in the predictor model need to be suitably parameterized by choosing a suitable model order from a set of possible candidate model orders based on a complexity criteria such as Akaike Information Criteria (AIC) or Bayesian Information Criteria (BIC) [27]. As the number of modules in the predictor model or the set of candidate model orders increase, the permutations to identify a suitable model order become computationally infeasible. In addition, parameterizing all the modules in the predictor model, results in an explosion of parameters which are not of primary interest to the experimenter and have a detrimental effect on the target module estimate.

In this study, we address two limitations of the existing methods. The first limitation is the reliance on availability of apriori topology information for building the predictor model. The second limitation is the need for a model order selection step and parameterization of all the modules in the predictor model to identify just one module.

For building a suitable predictor model, all the nodes that contain information about the output node are necessary. Such a set of nodes that contain all the information about a node is called the Markov blanket [30] of that node. However, this Markov blanket is defined for Bayesian networks (or Directed Acyclic Graphs (DAG)) which do not deal with confounding variables. Taking inspiration from the Markov blanket, we define the set of nodes that constitute the *Locality* of the output node for cyclic networks with confounding variables. Since the locality contains all the information about the output node, deleting the other nodes from the network does not affect the target module estimation. Deleting these other nodes also reduces the size of the network which has clear computational advantages. To identify the locality of the output node, we extend the non-causal Wiener filtering theory presented in [14] to cyclic networks with confounding variables. The smaller network obtained after deleting the nodes that do not belong in the locality of the output node is referred to as the local network. The nodal and noise interconnection information of the local network (referred to as the local topology) is necessary to build the predictor model.

The local topology is identified by searching for a candidate structure that minimizes a cost function. This cost function acts as a measure of closeness between the true topology and the candidate structure. The ideal structure is considered to be the one that minimizes this cost function. In a way, the topology identification problem is a problem of selecting an ideal model from all the candidate models. Inspecting all possible structure combinations would be computationally infeasible as these combinations increase exponentially with the size of locality. So a new algorithm is developed to search the ideal structure in lesser attempts. Moreover, we represent the candidate structure using non-parametric FIR models which results in a convex optimization problem. As a result, the local topology is identified through a series of convex optimizations with analytical solutions.

Based on the identified local topology, the predictor model is built using the algorithm for *Full Input Case* of local direct method [26] for effective estimation of target module. The obtained predictor model can be MISO or MIMO based on the identified topology. Based on the MISO/MIMO structure, the *Empirical Bayes Direct Method* [25] (for MISO) or the *Empirical Bayes Local Direct Method* [31] (for MIMO) is used to estimate the target module while simplifying the model order selection and reducing the nuisance parameters.

This paper is organized as follows. In section II, the system setup and the problem is defined. In section III, the method for identifying the locality is described. In section IV, the method for identifying the local topology is provided. In section V, the methods for building and solving the predictor model are discussed, including a complete algorithm for identifying the target module from scratch. In section VI, the numerical simulations and the results of the algorithm are presented, followed by discussions and conclusion in sections VII and VIII respectively.

II. System setup

A. Notation

Given a set $\mathcal{X}$, let $|\mathcal{X}|$ denote its cardinality.

B. Dynamic network setup

In this study we follow the dynamic network setting of [19]. To this end, we consider a network of L scalar *internal variables* (referred to as nodes) with dynamics defined by (1).

$$w_j(t) = \sum_{\substack{l=1 \\ l\neq j}}^{L} G_{jl}(z) w_l(t) + r_j(t) + v_j(t) \tag{1}$$

where, $j \in \{1, \ldots, L\}$ and z^{-1} is the delay operator.

- G_{jl} are stable, strictly proper, rational transfer functions with no poles on the unit circle and are referred to as *modules*,
- There are no self loops in the system i.e. nodes are not directly related to itself, $G_{jj} = 0$,
- $(\mathcal{I} - G(z))^{-1}$ exists, is proper and stable.
- $v_j(t)$ is the process noise acting on the node j, and
- $r_j(t)$ is the *external variable* that can be directly manipulated by the user. It may be absent in some nodes.

Collecting (1) for all the nodes, we get the expression (2)[2].

[2]time and frequency dependencies are dropped for convenience

$$\begin{bmatrix} w_1 \\ \vdots \\ w_L \end{bmatrix} = \begin{bmatrix} 0 & G_{12} \dots & G_{1L} \\ G_{21} & 0 & \dots & G_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ G_{L1} & G_{L2} \dots & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_L \end{bmatrix} + \begin{bmatrix} r_1 \\ \vdots \\ r_L \end{bmatrix} + \begin{bmatrix} v_1 \\ \vdots \\ v_L \end{bmatrix}. \tag{2}$$

Written as a matrix equation we get (3).

$$w(t) = G(z)w(t) + r(t) + v(t) \tag{3}$$

In this study, we consider only non-invasive observations (external variable $r(t) = 0$) since conducting an experiment by manipulating $r(t)$ increases the experiment cost.

The process noise $v(t)$ is modelled as a stationary stochastic process with rational spectral density $\Phi_v(\omega)$, such that there exists a white noise process $e(t)$ with covariance, $\Lambda > 0$, such that $v(t) = H(z)e(t)$, where H is a stable, stably invertible, monic and minimum phase transfer function matrix. The white noise process $e(t)$ is referred to as the innovation signal of the network (G, H). The process noise $v(t)$ may be correlated resulting in non-zero entries along the off-diagonal terms in the power spectral density $\Phi_v(\omega)$ and the noise model $H(z)$. This situation is referred to as a *dynamic network with correlated noise*.

C. Problem definition

In this study, we address the parametric identification of a single module connecting node i to node j i.e. G_{ji} in a dynamic network when no prior information is available about the network and noise topology, using only N measurement samples of $w_k(t)$ where $k = 1 \ \dots \ L$.

The predictor model for identifying this single module is built based on the network and noise interconnection structure of the set of nodes that contain all the information necessary for describing the output node. Before we identify the nodal and noise interconnection structure, we need to identify the nodes that contain information about the output node. In this study, we refer to such a set of nodes as the locality of the output node. The locality of a node can be seen as the Markov blanket [30] of the node in a network with correlated process noise. The Markov blanket of a node is the set of nodes that contain all the information about the node and is defined for Bayesian networks. Similarly, we need to define *locality* for networks with correlated process noise. To this end, we define the following sets to assist in defining the locality of the output node j.

Definition 1 (Children and Parents): Given a dynamic network (G, H), the children of a node j are defined by, $\mathcal{C}_j := \{i | G_{ij} \neq 0\}$. Similarly, the parents of the node j are defined by $\mathcal{P}_j := \{i | G_{ji} \neq 0\}$.

Definition 2 (Noise Confounders): Given a dynamic network (G, H), the noise confounders of a node j are defined by, $\mathcal{V}_j := \{i | \Phi_{v_j v_i} \neq 0\}$.

Note that the set $\mathcal{V}_j$ is symmetric, in a sense that, if $i \in \mathcal{V}_j$ then $j \in \mathcal{V}_i$. Also note that the set $\mathcal{V}_j$ contains the node j. Usual practice in literature is to represent the topology of the network by an adjacency matrix [19]. However, since in this study, we identify both the network and the noise topology, we formally redefine them as follows.

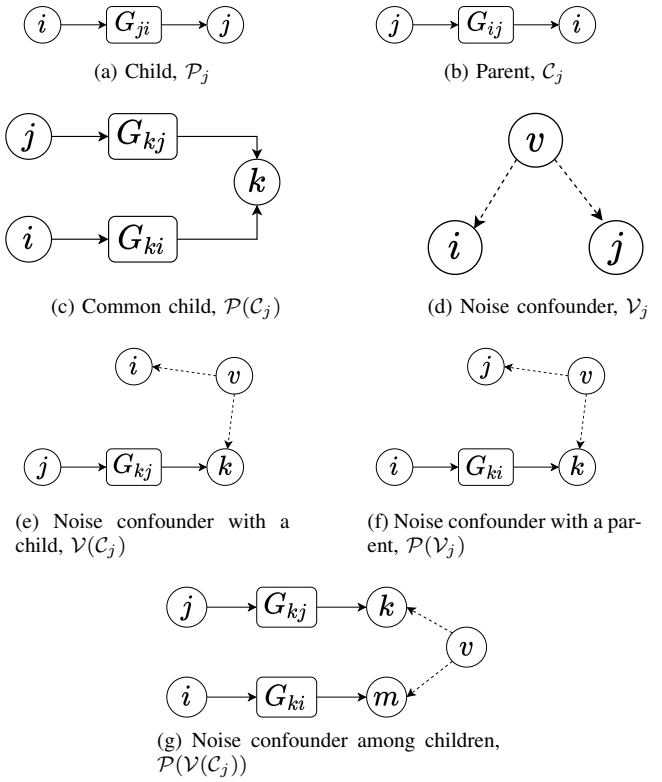


Fig. 1: Visual representation of the relationship between nodes i and j for the node i to be in the locality of j

Definition 3 (Topology): Consider a dynamic network (G, H), the topology of the network G, denoted by $\mathcal{T}_G$, is defined as follows,

$$\mathcal{T}_G(j, i) = \begin{cases} 1, & G_{ji} \neq 0 \\ 0, & G_{ji} = 0 \end{cases}.$$

The topology of the noise H, denoted by $\mathcal{T}_H$, is defined as follows,

$$\mathcal{T}_H(j, i) = \begin{cases} 1, & H_{ji} \neq 0 \\ 0, & H_{ji} = 0 \end{cases}.$$

With the help of the defined quantities, we now define the locality of the output node j.

Definition 4 (Locality of a node): Let j be a node in (3). The locality of the node j is denoted by $\mathcal{B}_j$. A node i belongs to the set $\mathcal{B}_j$ if it satisfies any of the following conditions

- $i = j$.
- $i \in \mathcal{P}_j$
- $i \in \mathcal{C}_j$
- $i \in \mathcal{P}(\mathcal{C}_j)$
- $i \in \mathcal{V}_j$
- $i \in \mathcal{V}(\mathcal{C}_j)$
- $i \in \mathcal{P}(\mathcal{V}_j)$
- $i \in \mathcal{P}(\mathcal{V}(\mathcal{C}_j))$.

Therefore, we define our first sub-problem as identifying the locality ($\mathcal{B}_j$) of j. Upon identifying the locality of the output node j, we delete the remaining nodes from the network. Based on the way we define the locality, it can be observed that deleting the remaining nodes in the network does not affect the output node or the interconnection structure of the output node. As a result, a smaller network is obtained which

is referred to as the local network and is denoted by $(\tilde{G}, \tilde{H})$. The process of removing a set of nodes and their signals from the network while preserving the second-order properties of the remaining nodes and signals is called immersion (refer to [28] for more information on immersion).

We define our second sub-problem as finding the network and noise topology of $(\tilde{G}, \tilde{H})$. We wish to do so by identifying $\mathcal{T}_{\tilde{G}}$, and $\mathcal{T}_{\tilde{H}}$ that a minimizes suitably selected cost function, through an iterative search algorithm.

The third sub-problem is defined as selecting the predictor inputs and predicted outputs using identified topology, $\hat{\mathcal{T}}_{\tilde{G}}$, and $\hat{\mathcal{T}}_{\tilde{H}}$ for consistent target module estimation. The selected predictor inputs and the predicted outputs constitute the predictor model.

The final sub-problem is defined as estimating a target module in a dynamic network that reduces mean square error, while simplifying the model order selection step and reducing the number of nuisance parameters in the predictor model. The formal problem definition of this study is defined in Problem 1.

Problem 1: With only N measurement samples of $w(t)$, how to find the target module estimate $\hat{G}_{ji}$ minimizing the mean square error, in a network (G, H) without any information on $\mathcal{T}_G$ and $\mathcal{T}_H$, simplifying the model order selection step and reducing the number of nuisance parameters.

III. Identifying the Locality

In this section, we develop a method to identify the locality of the output node in order to reduce the size of the network. Locality of the output node, as stated earlier, is synonymous to the Markov blanket defined for Bayesian Networks. A Markov blanket of a node j contains the Parents of the node j $(\mathcal{P}_j)$, Children of the node j $(\mathcal{C}_j)$ and other Parents of the Children of the node j $(\mathcal{P}(\mathcal{C}_j))$. It has been shown in [14] that a multivariate non-causal Wiener filter computed by projecting the node j on to the remaining nodal signals detect the Markov blanket of the corresponding node. Here, the sparsity of the Wiener filter is shown to coincide with the Markov blanket. However, this sparsity result was developed for networks without a noise model. In this section, we extend the results of [14] to networks with non-diagonal noise model, thereby proving that a multivariate non-causal Wiener filter of a node is capable of detecting the locality of that node. Following the theoretical results, we discuss the implementation of the non-causal Wiener filter.

A. Sparsity properties of non-causal Wiener filter

In order to identify the locality of j, we exploit the theory of Wiener filtering [32]. We derive conditions for the sparsity of non-causal Wiener filter obtained by projecting the node j on the remaining nodes in the network. Before deriving the conditions, we need to define the vector space containing the signals in $w(t)$. We define the following quantities to assist the vector space definition.

Definition 5 (Rationally related process [14]): Let $\mathcal{E}$ be a set containing time-discrete scalar, zero-mean, wide sense stationary processes such that for any $e_i, e_j \in \mathcal{E}$, the power spectral density $\Phi_{e_i e_j}(z)$ exists, is rational with no poles on the unit circle and is given by

$$\Phi_{e_i e_j}(z) = \frac{A(z)}{B(z)}$$

where, $A(z)$ and $B(z)$ are polynomials with real coefficients such that $B(z) \neq 0$ for any $|z| = 1$. Then, $\mathcal{E}$ is a set of rationally related process.

Definition 6 ([14]): The set $\mathcal{F}$ is defined as the set of rational single-input single-output (SISO) transfer functions that are analytic on the unit circle $\{z \in \mathbb{C} |\ |z| = 1\}$.

Definition 7 ([14]): Let $\mathcal{E}$ be a set of rationally related processes. Then, $\mathcal{FE}$ is defined as

$$\mathcal{FE} := \Big\{ w = \sum_{k=1}^{m} \Omega_k(z) e_k | e_k \in \mathcal{E},\ \Omega_k(z) \in \mathcal{F},\ m \in \mathbb{N} \Big\}.$$

Therefore, for any two processes $w_i, w_j \in \mathcal{FE}$,

$$\langle w_i, w_j \rangle := R_{w_i w_j}(0) = \int_{-\pi}^{\pi} \Phi_{w_i w_j}(e^{i\omega}) d\omega$$

defines an inner product for the vector space $\mathcal{FE}$.

Definition 8 ([14]): For a finite number of elements $w_1, \ldots, w_k \in \mathcal{FE}$, $tf - span$ is defined by,

$$tf - span\{w_1, \ldots, w_k\} := \Big\{ w = \sum_{i=1}^{k} \Omega_i(z) w_i | \Omega_i(z) \in \mathcal{F} \Big\}$$

Lemma 1 ([14]): The $tf - span$ operator defines a subspace of $\mathcal{FE}$.

Lemma 2: The nodal signals in $w(t)$ in (3) belong to the vector space $\mathcal{FE}$.

Proof: See Appendix I ■

With the vector space containing the signals in $w(t)$ defined, we now proceed to the formulation of non-causal Wiener filter for the defined space $\mathcal{FE}$ in the following proposition.

Proposition 1: Let y and $x_1, \ldots, x_n$ be processes in the space $\mathcal{FE}$. Define $x := (x_1, \ldots, x_n)^T$ and $X := tf - span\{x_1, \ldots, x_n\}$. Consider the problem

$$\inf_{q \in X} \|y - q\|^2$$

If $\Phi_x\left(e^{i\omega}\right) > 0$, for $\omega \in [-\pi, \pi]$, then the solution $\hat{y} \in X$ exists, is unique and is given by $\hat{y} = W(z)x$ where,

$$W(z) = \Phi_{yx}(z)\Phi_x(z)^{-1}. \tag{4}$$

The solution to (4) is non-causal and $W(z)$ is the non-causal Wiener filter. Moreover, $\hat{y}$ is the only element in X such that for any $q \in X$,

$$\langle y - \hat{y}, q \rangle = 0 \tag{5}$$

Proof: Refer to [14] for proof. ■

Note that (5) represents the Wiener filter orthogonality condition. This orthogonality condition is used in deriving the sparsity conditions of the non-causal Wiener filter and in the implementation of the Wiener filter. With the non-causal Wiener filter formulated, we define the following theorem that provides sufficient conditions for categorizing a node in the locality of another.

Theorem 1 (Sparsity of the non-causal Wiener filter): Define $w := \left(w_1, \ldots, w_L\right)^\top$ to be the nodal signals obtained

Algorithm 1 Identifying the locality ($\mathcal{B}_j$) of a node j

Input:$\{w_k(t)\}_{t=1}^{N}, k \in \{1,\dots,L\}$
Output: $\mathcal{B}_j$

1) Set $\bar{w} = \{w_k\}_{k\neq j}$.
2) Compute a multivariate, non-causal $W_j(z)$ with j as output and $\bar{w}$ as input (refer to Appendix III).
3) Initialize $\mathcal{B}_j = \emptyset$
4) **for** $i = 1 : L \setminus j$
   a) **if** $\|W_{jk}\|_\infty > \rho$, add k to $\mathcal{B}_j$
   b) **continue**

from (3). Define $\bar{w} := \left(w_1,\dots,w_{j-1},w_{j+1},\dots,w_L\right)^\top$ and the space $\mathcal{W}_j := tf - span\{\bar{w}\}$. Consider the problem of approximating w_j with an element $\hat{w}_j \in \mathcal{W}_j$ (non-causal Wiener filtering) defined as follows:

$$\min_{\hat{w}_j\in\mathcal{W}_j} \|w_j - \hat{w}_j\|^2. \tag{6}$$

Then, the optimal solution $\hat{w}_j$ exists, is unique, and is given by

$$\hat{w}_j = \sum_{i\neq j} W_{ji}(z)w_i = W_j(z)\bar{w}, \tag{7}$$

where, $W_{ji}(z) \neq 0$ implies $w_i \in \mathcal{B}_j$

Proof: See Appendix II ∎

From theorem 1, it can be seen that a node must be in the locality of node j to obtain a non-zero Wiener filter entry. Therefore, to identify the locality of j, we simply need to compute the multivariate non-causal Wiener filter by projecting the node j on the remaining nodes in the network and group the nodes that have a non-zero Wiener filter entry. These nodes belong in the locality of node j. With this, we discuss the implementation of the Wiener filter in the following subsection.

B. Implementation of non-causal Wiener filter

The theoretical results presented in the previous subsection rely on the detection of the non-zero links in the $W_{ji}(z)$ i.e. $W_{ji} \neq 0$. However, the Wiener filter entries in practical implementation and formulation cannot be exactly zero. Therefore, a threshold ρ needs to be defined such that $\|W_{ji}\|_\infty > \rho$ implies that the corresponding Wiener filter entry is non-zero. The value of ρ can be chosen based on the signals w_j and w_i.

For the implementation of the non-causal Wiener filter, we consider the non-causal impulse response representation of the Wiener filter followed in [33]. Here, the infinite impulse response is truncated and an approximated FIR Wiener filter of length F is generated to study the sparsity. We choose this method of implementation due to its ease of computation. However, opting for a different implementation method does not affect the results of the Wiener filter. The algorithm for identifying the $\mathcal{B}_j$ is given in Algorithm 1.

IV. Identifying the local topology

In this section, we develop a method to identify the network and noise topology $(\mathcal{T}_{\tilde{G}}, \mathcal{T}_{\tilde{H}})$ of the network $(\tilde{G}, \tilde{H})$ obtained after immersing the nodes that are absent in the locality of the output node. This immersed network is referred to as the local network.

Identifying the network topology is a straightforward problem for which numerous solutions are provided in literature. However, identifying the noise topology is a more complicated problem as the innovation signal necessary for identifying the noise structure is not readily available. For this reason, we follow the approach presented in [34] where the innovation signal is estimated initially using a non-parametric model.

Similar to [34], the first step in the local topology identification is to estimate the innovation using a non-parametric model. Upon obtaining the innovation estimate, we treat these estimates as additional nodal signals and simultaneously estimate the network and noise topology. We discuss the details of the developed method in the following subsections.

A. Innovation estimation

In this subsection, we discuss the estimation of the innovation signal in the local network $(\tilde{G}, \tilde{H})$. To this end, we first define the local network and its dynamics obtained after immersion in (8).

$$w_{\mathcal{B}}(t) = \tilde{G}w_{\mathcal{B}}(t) + \tilde{H}\xi_{\mathcal{B}}(t). \tag{8}$$

In (8), $w_{\mathcal{B}}$ are the nodal signals of $\mathcal{B}_j$, $\tilde{G}$ is a hollow, strictly proper transfer function matrix, and $\tilde{H}\xi_{\mathcal{B}}(t)$ is obtained by spectral factorization where $\xi_{\mathcal{B}}(t)$ is a white noise vector (see [26] for more details). To estimate the innovation $\xi_{\mathcal{B}}(t)$, we use non-parametric high-order ARX modelling for reasons of convexity and computational convenience. To justify this choice of modeling, we define the following proposition to show that (8) can be represented in an ARX form.

Proposition 2: For every dynamic network given by (8), there exists a network representation,

$$Qw_{\mathcal{B}}(t) = Pw_{\mathcal{B}}(t) + \xi_{\mathcal{B}}(t), \tag{9}$$

where, Q is a diagonal, and monic transfer function matrix, and P is a hollow, strictly proper transfer function matrix.

Proof: See Appendix IV ∎

Every module present in Q and P is represented as an independent impulse response,

$$Q_{mm} = 1 + \sum_{d=1}^{\infty} q_d^m z^{-d} \approx 1 + \sum_{d=1}^{\ell_{ARX}} q_d^m z^{-d} \tag{10}$$

$$P_{mn} = \sum_{d=1}^{\infty} p_d^{mn} z^{-d} \approx \sum_{d=1}^{\ell_{ARX}} p_d^{mn} z^{-d}. \tag{11}$$

All impulse responses are approximated to a length ℓ_{ARX} chosen sufficiently long to capture all the dynamics in the local network. This style of representation allows us to estimate the innovation in a much simpler manner. By representing the local network as (9), each row of Q and P can be identified independently in a parallel MISO setup. Therefore, the linear regressor form of a node $m \in \mathcal{B}_j$ can be written as

$$w_m(t) = \varphi_m^\top(t)\eta_{ARX}^m + \xi_m(t). \tag{12}$$

Here, $\eta^m_{ARX} = \begin{bmatrix} p^{mn_1} \dots p^{mn_{m-1}} \, q^m \, p^{mn_{m+1}} \dots p^{mn_{|\mathcal{B}_j|}} \end{bmatrix}^\top$, and $\varphi^\top_m(t) = \begin{bmatrix} \varphi^{n_1}_m(t) & \dots & \varphi^{n_{|\mathcal{B}_j|}}_m(t) \end{bmatrix}$, with $p^{mn} = \begin{bmatrix} p^{mn}_1 \dots p^{mn}_{\ell_{ARX}} \end{bmatrix}$, $q^m = \begin{bmatrix} -q^m_1 \dots -q^m_{\ell_{ARX}} \end{bmatrix}$, and $\varphi^n_m(t) = \begin{bmatrix} w_n(t-1) & \dots & w_n(t-\ell_{ARX}) \end{bmatrix}$, where $n_1, \dots, n_{|\mathcal{B}_j|}$ are elements of $\mathcal{B}_j$ and $n \in \mathcal{B}_j$.

We parametrize all the elements of η^m_{ARX}, resulting in the following one-step ahead predictor [27],

$$\hat{w}_m = \varphi^\top_m(t)\eta^m_{ARX}. \tag{13}$$

We estimate the parameter vector $\hat{\eta}^m_{ARX}$ by minimizing the following identification criterion

$$\hat{\eta}^m_{ARX} = \underset{\eta^m_{ARX}}{\operatorname{argmin}} \sum_{t=1}^{N} \epsilon^2_m(t, \eta^m_{ARX}) + (\eta^m_{ARX})^\top R^m \eta^m_{ARX} \tag{14}$$

where, $\epsilon_m(t, \eta^m_{ARX}) = w_m(t) - \varphi^\top_m(t)\eta^m_{ARX}$ is the prediction error, and $R^m = \operatorname{diag}(R^m_1, \dots, R^m_{|\mathcal{B}_j|})$ is the regularization matrix to handle the excessive variance of the estimate. The regularization term R^m_k for $k \in \mathcal{B}_j$ is chosen to be a modified Tuned/Correlated (TC) kernel [35] as it enforces stability. The modified TC kernel has the following structure

$$R^m_k = \operatorname{diag}(1, \frac{1}{\alpha_k}, \dots, \frac{1}{\alpha^{\ell_{ARX}-1}_k}). \tag{15}$$

Here, the α_k represents the decay rate of the impulse response of the corresponding module and is estimated by cross-validation [35]. Due to the convexity of ARX modeling, $\hat{\eta}^m_{ARX}$ has a closed form expression given below

$$\hat{\eta}^m_{ARX} = \left[\frac{1}{N} \sum_{t=1}^{N} \varphi_m(t)\varphi^\top_m(t) + R^m \right]^{-1} \cdot \frac{1}{N} \sum_{t=1}^{N} \varphi_m(t) w_m(t). \tag{16}$$

With an estimate of $\hat{\eta}^m_{ARX}$, we obtain the prediction error $\epsilon_m(t, \hat{\eta}^m_{ARX})$. The prediction error estimated using (14), $\epsilon_m(t, \hat{\eta}^m_{ARX})$, acts as the estimate of the innovation $\hat{\xi}_m(t)$. Having obtained estimates of all innovations in the local network in a similar manner, we now proceed to identifying the local topology.

B. Structure selection

In this subsection, we discuss the method for identifying the local topology $(\mathcal{T}_{\tilde{G}}, \mathcal{T}_{\tilde{H}})$ from the signals $(w_{\mathcal{B}}, \hat{\xi}_{\mathcal{B}})$. To this effect, we re-write the dynamics of the node $m \in \mathcal{B}_j$ as

$$w_m(t) = \sum_{n \in \mathcal{P}_m} \tilde{G}_{mn} w_n(t) + \sum_{n \in \mathcal{V}_m} \tilde{H}_{mn} \xi_n(t) + \xi_m(t), \tag{17}$$

where modules are represented as impulse response models as shown in (18).

$$\tilde{H}_{mn} = \sum_{d=1}^{\infty} h^{mn}_d z^{-d}, \; \tilde{G}_{mn} = \sum_{d=1}^{\infty} g^{mn}_d z^{-d}. \tag{18}$$

By rewriting the dynamics as (17), we can observe the effect $\hat{\xi}_{\mathcal{B}}$ has on $w_m(t)$. The noise confounders of the node m are clearly seen as inputs alongside the parents of the node m. Therefore, with an estimate of $\hat{\xi}_{\mathcal{B}}$ available, we can build a predictor model using both the nodal signals and innovation signals as predictor inputs. Since we represent the modules as impulse response models, we can write the linear regressor form of node m as

$$w_m(t) = \bar{\varphi}^\top_m(t)\eta^m_{FIR} + \xi_m(t), \tag{19}$$

where, $\bar{\varphi}_m$ contains time shifted signals (similar to (12)) of the parents and noise confounders of the node m, and η^m_{FIR} are the parameters of the corresponding impulse response model.

After defining the impulse response representation of the local network, we advance to the simultaneous estimation of η^m_{FIR} and m^{th} row of $(\mathcal{T}_{\tilde{G}}, \mathcal{T}_{\tilde{H}})$. To identify the local topology, we need to define a criterion to distinguish between two different candidate structures on the basis of data. The defined criterion should be an indirect measure of how close a certain candidate structure is to the true structure given in (17).

In this regard, we consider a modified representation of the *Akaike Information Criterion (AIC)* and *Bayesian Information Criterion (BIC)* mentioned in [27] as shown in (20).

$$\begin{aligned} J_{AIC}(\hat{\eta}^m_{FIR}) &= \log\left(\frac{1}{N} \sum_{t=1}^{N} \epsilon^2(t, \hat{\eta}^m_{FIR}) \right) + \frac{2 N_G \ell_{FIR}}{N} \\ J_{BIC}(\hat{\eta}^m_{FIR}) &= \log\left(\frac{1}{N} \sum_{t=1}^{N} \epsilon^2(t, \hat{\eta}^m_{FIR}) \right) + \frac{N_G \ell_{FIR}}{N} \log(N) \end{aligned} \tag{20}$$

Here, N_G is the total number of nodes and innovations considered as predictor inputs in the candidate structure, ℓ_{FIR} is the length of the impulse responses in (18) and is chosen long enough to sufficiently represent the dynamics of the network. The choice of BIC is motivated by its consistency in identifying the true model with increasing data [36], which is not the case for AIC. However, the convergence rate of AIC is higher than BIC, making it a better choice at smaller sample sizes [37]. Having defined the selection criteria, we estimate $\hat{\eta}^m_{FIR}$ in an iterative manner for different choices of predictor inputs.

Let $\mathcal{W}_m$ and $\mathcal{H}_m$ be the set of nodal signals and innovation signals considered as predictor inputs in the candidate structure. The corresponding one-step ahead predictor is as follows

$$\hat{w}_m(t) = \bar{\varphi}^\top_m(t)\eta^m_{FIR}. \tag{21}$$

Here, $\eta^m_{FIR} = \begin{bmatrix} g^{mn_1} \dots g^{mn_{|\mathcal{W}_m|}} \, h^{mk_1} \dots h^{mk_{|\mathcal{H}_m|}} \end{bmatrix}^\top$, and $\bar{\varphi}^\top_m(t) = \begin{bmatrix} \varphi^{n_1}_m(t) \dots \varphi^{n_{|\mathcal{W}_m|}}_m(t) \, \varphi^{k_1}_m(t) \dots \varphi^{k_{|\mathcal{H}_m|}}_m(t) \end{bmatrix}$, with $g^{mn} = \begin{bmatrix} g^{mn}_1 \dots g^{mn}_{\ell_{FIR}} \end{bmatrix}$, $h^{mk} = \begin{bmatrix} h^{mk}_1 \dots h^{mk}_{\ell_{FIR}} \end{bmatrix}$, $\varphi^n_m(t) = \begin{bmatrix} w_n(t-1) & \dots & w_n(t-\ell_{FIR}) \end{bmatrix}$, and $\varphi^k_m(t) = \begin{bmatrix} \hat{\xi}_k(t-1) & \dots & \hat{\xi}_k(t-\ell_{FIR}) \end{bmatrix}$, where $n_1, \dots, n_{|\mathcal{W}_m|}$ are the elements of $\mathcal{W}_m$, $k_1, \dots, k_{|\mathcal{H}_m|}$ are the elements of $\mathcal{H}_m$, $n \in \mathcal{W}_m$, and $k \in \mathcal{H}_m$.

The identification criterion for this candidate structure is as follows

$$\hat{\eta}^m_{FIR} = \underset{\eta^m_{FIR}}{\operatorname{argmin}} \sum_{t=1}^{N} \epsilon^2_m(t, \eta^m_{FIR}). \tag{22}$$

where, $\epsilon_m(t, \eta^m_{FIR}) = w_m(t) - \hat{w}_m(t)$. The estimate of $\hat{\eta}^m_{FIR}$

is a closed form expression as follows

$$\hat{\eta}_{FIR}^{m} = \left[\frac{1}{N}\sum_{t=1}^{N}\bar{\varphi}_m(t)\bar{\varphi}_m^{\top}(t)\right]^{-1} \cdot \frac{1}{N}\sum_{t=1}^{N}\bar{\varphi}_m(t)w_m(t). \quad (23)$$

The estimate $\hat{\eta}_{FIR}^{m}$ obtained from (23) is used to compute $\epsilon_m(t, \hat{\eta}_{FIR}^{m})$ which in turn is used to compute the cost of the selection criterion for this candidate structure. To find the ideal structure that minimizes the selection criterion, a total $2^{|\mathcal{B}_j|}$ combinations need to be tested out. Testing these combinations become computationally infeasible as $|\mathcal{B}_j|$ increases. To reduce the number of combinations to test, we develop an iterative search algorithm (referred to as the Focus search algorithm) that is greedy in its approach to identify the ideal structure.

Consider a node $m \in \mathcal{B}_j$, whose interconnection structure we wish to identify. In the absence of interconnection information, it is safe to assume that $\hat{\xi}_m$ is connected to the node m. Therefore, in the focus search algorithm, we initialize the structure with the innovation signal $\hat{\xi}_m$ and compute the cost of the selection criterion for this structure. With this computed cost as the base cost, we search for one nodal or innovation signal that minimizes the selection criterion. This nodal/innovation signal is then added to the structure and the corresponding cost becomes the new base cost. This step is repeated with the remaining nodal and innovation signals until the cost saturates. The interconnection information is obtained from the resulting structure. The focus search algorithm along with the remaining steps for identifying $(\mathcal{T}_{\tilde{G}}, \mathcal{T}_{\tilde{H}})$ are presented in Algorithm 2.

Algorithm 2 Identification of local topology $(\mathcal{T}_{\tilde{G}}, \mathcal{T}_{\tilde{H}})$

Input:$\{w_k(t)\}_{t=1}^{N}, k \in \mathcal{B}_j$

Output: $\hat{\mathcal{T}}_{\tilde{G}}, \hat{\mathcal{T}}_{\tilde{H}}$

1) **for** $m \in \mathcal{B}_j$
   a) Estimate the α_k, $k \in \mathcal{B}_j$ by cross validation (refer to [35] for details).
   b) Identify $\hat{\eta}_{ARX}^{m}$ using (16).
   c) Generate $\hat{\xi}_k(t, \hat{\eta}_{ARX}^{m})$ using (14).
2) **end for**
3) Choose an appropriate selection criteria.
4) **for** $m \in \mathcal{B}_j$
   a) Initialize $\mathcal{W}_m = \emptyset$, and $\mathcal{H}_m = \{\hat{\xi}_m\}$.
   b) Estimate $\hat{\eta}_{FIR}^{m}$ using (23).
   c) Compute the J_{cost} using (20).
   d) **for** $count = 1 : 2|\mathcal{B}_j|^2 - |\mathcal{B}_j|$
      i) **for** $k \in \mathcal{B}_j$
         A) Set $\mathcal{W}_m = \{w_k\}$.
         B) Estimate $\hat{\eta}_{FIR}^{m}$ using (23).
         C) Compute $J_{iteration}$ using (20).
         D) **if** $J_{iteration} < J_{base}$
         E) Set $J_{base} = J_{iteration}$.
         F) Set $\mathcal{S}_m^{temp} = w_k$
         G) **else**
         H) **continue**
      ii) **for** $k \in \mathcal{B}_j \setminus m$
         A) Set $\mathcal{H}_m = \{\hat{\xi}_k\}$.
         B) Estimate $\hat{\eta}_{FIR}^{m}$ using (23).
         C) Compute $J_{iteration}$ using (20).
         D) **if** $J_{iteration} < J_{base}$
         E) Set $J_{base} = J_{iteration}$.
         F) Set $\mathcal{S}_m^{temp} = \hat{\xi}_k$
         G) **else** continue
         H) **end if**
      iii) **end for**
      iv) Add $\mathcal{S}_m^{temp}$ to $\mathcal{W}_m$ or $\mathcal{H}_m$ accordingly.
   e) **end for**
   f) Set the non-zero entries for m^{th} row of $\hat{\mathcal{T}}_{\tilde{G}}, \hat{\mathcal{T}}_{\tilde{H}}$ based on $\mathcal{W}_m$ and $\mathcal{H}_m$.
5) **end for**

V. Identifying the target module

In this section, we present the identification of the target module based on the identified local topology $(\hat{\mathcal{T}}_{\tilde{G}}, \hat{\mathcal{T}}_{\tilde{H}})$. To do so, there are two crucial steps, namely forming the predictor model for effective identification of the target module, and obtaining an efficient estimate of the target module. A predictor model consists of predictor input signals and predicted output signals, which are chosen suitably to avoid a possible bias in parameter estimation. To suitably select the signals in this predictor model, we use the algorithms presented in the *Local Direct method* [26]. The desire not to depend on the presence of external signals, the maximum likelihood properties of the direct methods and the ability to handle correlated process noise make the local direct method a suitable choice for building the predictor model.

According to the local direct method, a module embedded in a dynamic network with correlated process noise can be consistently identified with an estimation setup of $w_{\mathcal{D}} \rightarrow w_{\mathcal{Y}}$, where $w_{\mathcal{D}}$ are the set of predictor input signals and $w_{\mathcal{Y}}$ are the set of predicted output signals. This method handles the effect of confounding variables in two approaches. The first approach is adding additional predictor inputs while the second approach is adding additional predicted outputs. As a result, the predictor model may have a MISO structure or a MIMO structure depending on the correlated noise. The local direct method provides three algorithms for selecting the predictor input signals and predicted output signals. The algorithm for *Full input case* of the local direct method is the most suitable for this study since this algorithm chooses the predictor model with the least number of outputs when access to all nodal signals are available.

The obtained predictor model can have a MISO structure or a MIMO structure, depending on the presence of correlated noise in the local network. Irrespective of the structure, to identify one target module, all the modules in the predictor model need to be suitably parameterized by a model order selection step. This step becomes computationally expensive as the number of modules in the predictor model increases. Also, parameterizing all the modules result in an explosion of nuisance parameters that affect the target module estimate. As a result, to circumvent model order selection and reduce the number of nuisance parameters, we consider the regular-

Algorithm 3 Target module identification for networks with unknown network and noise topology

Input:$\{w_k(t)\}_{t=1}^{N}, k \in \{1,\dots,L\}$

Output: $\hat{\theta}$

1) Identify the $\mathcal{B}_j$ using Algorithm 1.
2) Identify the local topology $(\mathcal{T}_{\tilde{G}}, \mathcal{T}_{\tilde{H}})$ using Algorithm 2.
3) Based on the identified local topology, formulate the predictor model using the algorithm for *Full Input case* in [26].
4) Use EBDM [25], or EBLDM [31] based on the structure of the predictor model and estimate the target module.

ized kernel based methods, *Empirical Bayes Direct Method* (EBDM) of [25] and *Empirical Bayes Local Direct Method* (EBLDM) of [31].

In both these methods, the problem of model order selection step is simplified by opting for a non-parametric impulse response modeling of all modules except the target module. As a result, the problem of model order selection gets simplified to choosing a sufficiently long impulse response. To reduce the number of parameters, each impulse response is modelled as a Gaussian Process (GP) governed by a *Stable spline (SS) kernel* [35]. As a result, an impulse response of any length can be captured using only two hyperparameters that govern the SS kernel. In these methods, the parameters are identified by maximizing the marginal likelihood of data which inherently minimizes the mean square error of the estimation problem. This further reduces the variance of estimated parameters of the target module.

Therefore, for a predictor model with MISO structure, EBDM is used to estimate the target module, while for a predictor model with MIMO structure, EBLDM is used to estimate the target module. With this, we present the final algorithm that solves the *Problem* 1 in Algorithm 3.

VI. Numerical simulation

To evaluate the performance of the algorithm, we perform numerical simulations. The developed algorithm can be split into two parts, identifying the local topology (Steps 1 and 2 of Algorithm 3) and identifying the target module (Steps 3 and 4 of Algorithm 3). We conduct two simulation experiments, the first to highlight the effectiveness of Algorithms 1 and 2 in identifying the topology, and the second to highlight the effectiveness of the complete Algorithm 3 in identifying the target module.

A. Simulation experiment 1

To highlight the effectiveness of Algorithms 1 and 2, we generate 50 random stable networks (modules of 2^{nd} order are randomly generated) with the network topology as shown in the Figure 2. In this 10-node network, the process noise (v_1, v_8), (v_4, v_5), and (v_7, v_9) are correlated and the noise structure is provided in (24).

$$v(t) = He(t) \tag{24}$$

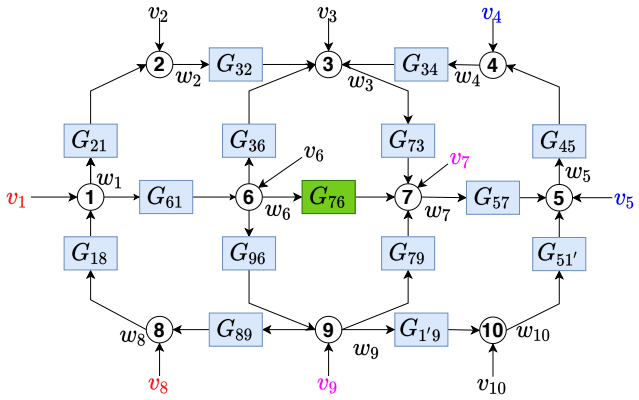


Fig. 2: 10-Node dynamic network with a process noise for each node. The process noise (v_1, v_8), (v_4, v_5), and (v_7, v_9) are correlated.

where, H is a monic, stable and stably invertible transfer function matrix with a monic diagonal and non-zero strictly proper entries in $H(1,8)$, $H(4,5)$, $H(5,4)$, $H(7,9)$, $H(8,1)$, and $H(9,7)$. The noise source $e(t)$ is a zero-mean gaussian process with covariance matrix $\Lambda = \mathcal{I}$ and is the only source of excitation in the network. We collect $N = 1000$, $N = 5000$ and $N = 10000$ measurement samples from all nodes in the network for all 50 networks.

In this network, the module that we wish to identify is G_{76}. Using Algorithm 1, we first identify the locality of the node 7, and immerse the remaining nodes. The Wiener filter length F is commonly chosen to be 20 across all the different networks but thresh ρ is chosen independently based on each network since the non-causal Wiener filter dynamics is different for all networks. The threshold value ρ is chosen after visually examining the obtained Wiener filters. Following this, we identify the local topology using Algorithm 2 with $\ell_{ARX} = 50$. The behaviour of this algorithm is studied by comparing the *True Positive Rate (TPR)* and the *False Positive Rate (FPR)* over different choices of ℓ_{FIR} and selection criteria. The expressions for TPR and FPR are as follows

$$\text{TPR} = \frac{\text{TP}}{\text{P}}, \quad \text{FPR} = \frac{\text{FP}}{\text{A}}.$$

Here, P refers to the number of positive interconnections in the network, TP refers to the number of instances of positive interconnection identified by the algorithm, A refers to the number of negative interconnections in the network, and FP refers to the number of instances of negative interconnections falsely identified by the algorithm. The developed search algorithm is also compared to the *forward-backward greedy search* algorithm of [38].

The obtained TPR vs FPR for different data lengths over different $\ell_{FIR} = (10, 20, 40, 80)$ is shown in the Figure 3. The quiver represent the direction of increasing ℓ_{FIR}. In the sub figures (a), (b) and (c) of Figure 3, the TPR vs FPR values for different selection criteria using different search algorithms are shown. The red curves represent AIC while the blue curves represent BIC. From the figures, it can be seen that the TPR vs FPR curves for AIC are on an average closer to $(0, 1)$ than the TPR vs FPR curves for BIC. This observation can be

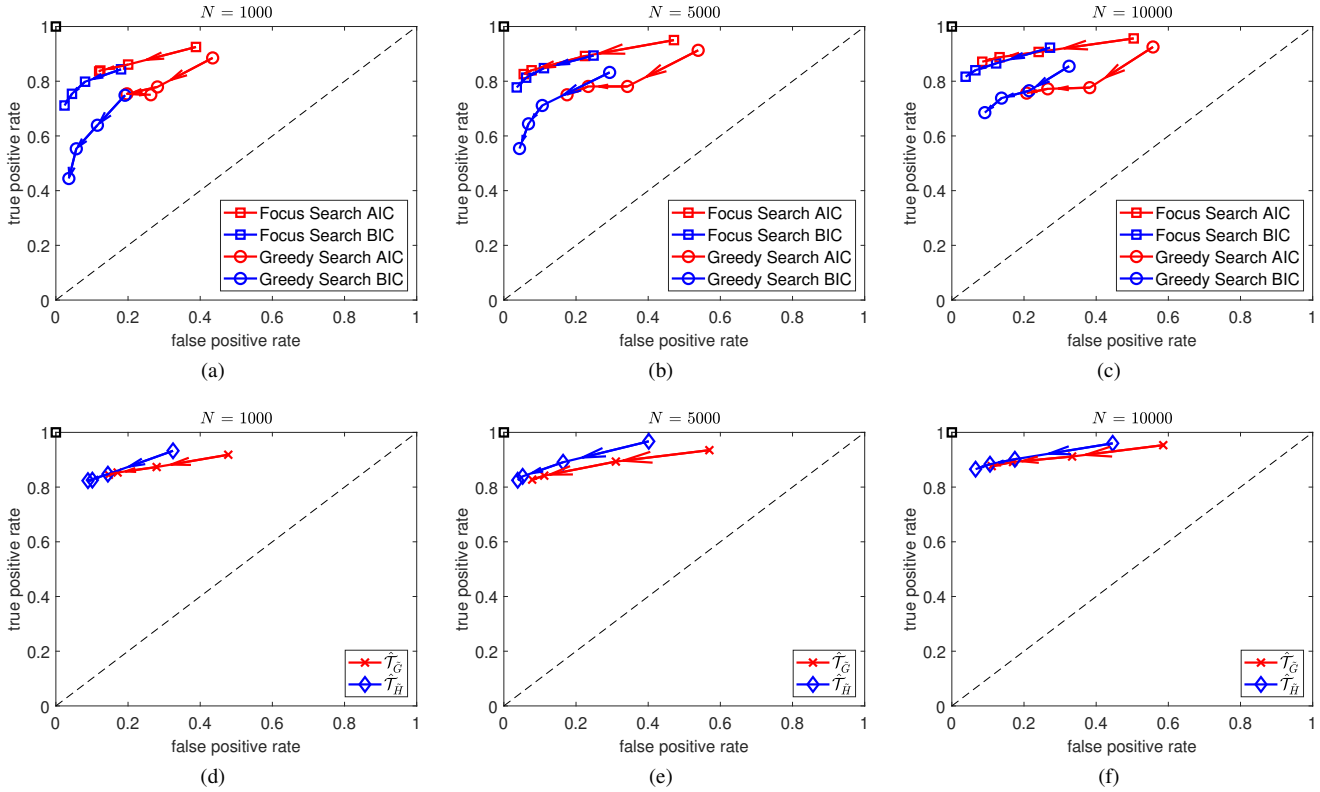


Fig. 3: True Positive Rate and False Positive Rate over different $\ell_{FIR} = (10, 20, 40, 80)$ for different data length, $N = 1000$ (left), $N = 5000$ (middle), $N = 10000$ (right). The quiver represents the direction of increasing ℓ_{FIR}. The sub figures (a)-(c) show the TPR vs FPR for different selection criteria and different search algorithms . The sub figures (d)-(f) show the TPR vs FPR for nodal topology ($\hat{\mathcal{T}}_{\tilde{G}}$) and noise topology ($\hat{\mathcal{T}}_{\tilde{H}}$) using AIC as selection criterion.

attributed to the faster convergence rate of AIC [37]. Another phenomenon that can be observed in these figures is that as N increases, the curves corresponding to BIC move closer to the $(0, 1)$ and its overlap with the curves corresponding to AIC increases. This is due to the slower convergence rate of BIC. However, since BIC has consistency, as the number of samples increases, the identified model converges to the true model.

In the sub figures (a), (b) and (c) of Figure 3, the curves with a square marker correspond to the focus search algorithm and the curves with a circle marker correspond to the forward-backward greedy search algorithm. It can be seen that the focus search has better values of TPR vs FPR than the greedy search algorithm.

A comparison of performance between the identified $\hat{\mathcal{T}}_{\tilde{G}}$ and $\hat{\mathcal{T}}_{\tilde{H}}$ using AIC as the selection criteria is shown in sub figures (d), (e) and (f) of Figure 3. It can be seen from these sub figures that as ℓ_{FIR} increases, the curves get closer to $(1, 0)$ until $\ell_{FIR} = 40$. Beyond $\ell_{FIR} = 40$, the curves do not get any closer to $(0, 1)$. This saturation is observed because the average impulse response length of the true network is approximately 40. Beyond this value, it can be observed that the performance of the algorithm is uniform in identifying the nodal and noise topology.

We also define a distance metric dis to quantify the closeness of the identified structure to the true structure as follows,

$$dis = \sqrt{(1 - TPR)^2 + FPR^2}.$$

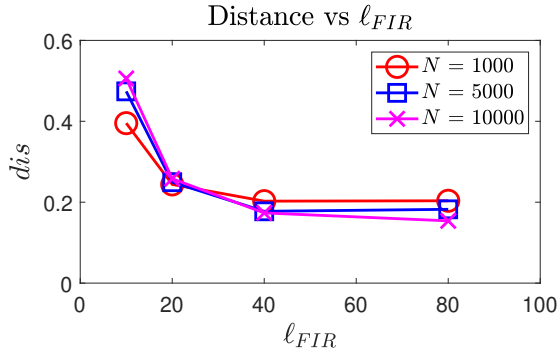


Fig. 4: Distance versus ℓ_{FIR} for different data lengths using AIC as selection criterion

This metric measures the distance of (FPR,TPR) from (0,1), and a smaller dis implies a better performance. The Figure 4 shows the distance vs ℓ_{FIR} comparison using AIC as selection criteria. From Figure 4, it can be observed that the distance reduces only until ℓ_{FIR} is large enough to capture the entire dynamics of the network.

B. Simulation experiment 2

To highlight the effectiveness of the complete Algorithm 3, we consider one of the 50 randomly generated networks for evaluating the performance of the complete algorithm

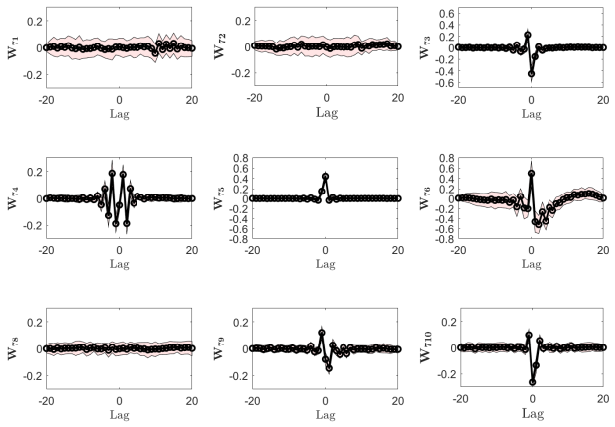


Fig. 5: Non-causal Wiener filter estimate $\hat{W}_7$ obtained by projecting w_7 on the remaining nodes in the network. The black line represents mean of the estimate over 50 MC simulations. The red shaded region represents the standard deviation of the estimates over 50 MC simulations.

from a parameter identification perspective. To prevent issues that might arise due to lack of sufficient excitation for target module identification, we add an additional white noise source with unit power on each node in the network. We run 50 independent Monte Carlo (MC) experiments for different realizations of the noise source $e(t)$ while keeping the dynamics of the network fixed. We collect $N = 1000$ samples of all the nodes for each MC simulation. The network dynamics for the MC simulations can be found in the Appendix V. For convenience, the target module is given below.

$$G_{76} = \frac{0.1050q^{-1} - 0.3465q^{-2}}{1 + 0.0480q^{-1} - 0.2534q^{-2}} = \frac{b_1q^{-1} - b_2q^{-2}}{1 + a_1q^{-1} + a_2q^{-2}} \tag{25}$$

We implement Algorithm 3 to identify the parameters. The non-causal Wiener filter length is chosen to be 20 while threshold ρ for identifying the locality is chosen to be 0.18. The true locality for this network is $\mathcal{B}_7 = \{3, 4, 5, 6, 9, 10\}$ and only this set of nodes should be non-causally Wiener correlated to the node 7. The mean and standard deviation plot of the non-causal Wiener filter between node 7 and the other nodes are shown in the Figure 5. Although in some iterations, a few nodes are falsely identified to be in the locality due to variance of the parameters, the mean of the filter (shown in black in Figure 5) over the 50 MC simulations preserve the sparsity conditions derived in Theorem 1.

For identifying the local topology, we choose $\ell_{ARX} = 50$ and $\ell_{FIR} = 40$ for the lengths of impulse response and AIC as the selection criteria. The performance of the algorithm for local topology estimation is shown in the table I.

To evaluate the performance of the identified topology in estimating the target module, we build predictor models based on the identified topology and true topology and compare the target module estimates obtained therein. The predictor model for identifying G_{76} based on the true topology is obtained using the *Full input case* of local direct method [26]. Therefore, the predictor inputs are chosen to be (w_6, w_3, w_9).

TABLE I: TPR, FPR and dis for the numerical simulation 2 with $\ell_{ARX} = 50$ and $\ell_{FIR} = 40$

	TPR	FPR	dis
Overall	0.8417	0.1175	0.1971
Node	0.8617	0.1548	0.2076
Noise	0.8200	0.0911	0.2017

Since the input w_9 has a process noise correlation with the output w_7, w_9 is added to the predicted outputs along with w_7. Therefore, the predictor model is $\{w_6, w_3, w_9\} \rightarrow \{w_7, w_9\}$. The predictor model based on the identified topology is also obtained using the same method. The identification method in step 4 of Algorithm 3 is chosen as EBLDM since the predictor model has a MIMO structure.

Laterally, we compare the performance of the EBLDM in its effectiveness to obtain a target module estimate, with the Direct method (DM) [19]. Therefore, the predictor model for the DM based on the true topology is $\{w_6, w_3, w_9\} \rightarrow \{w_7\}$ [19]. The predictor model for the DM based on the identified topology is also obtained using the same method.

In addition to the classic DM, the innovation estimate $\hat{\xi}_i$ available at our disposal can help improve the target module estimation. In section IV, we use the innovation estimate $\hat{\xi}_i$ in the predictor model to remove the effect of confounding variables. Following the same approach here, we incorporate the innovation estimates $\hat{\xi}_i(t)$ as predictor inputs in an attempt to reduce the confounding effect of correlated noise and improve the target module estimation. Therefore, the predictor model for this estimation problem based on the true topology is $\{w_6, w_3, w_9, \hat{\xi}_7, \hat{\xi}_9\} \rightarrow \{w_7\}$. We follow the same procedure for building the predictor model based on identified topology. We estimate these MISO predictor models using the DM and EBDM.

Therefore, we compare the EBLDM with predictor models obtained from true topology (referred to as 'EBLDM+TT') and identified topology (referred to as 'EBLDM+IT') to the DM with predictor models obtained from true topology (referred to as 'DM+TT') and identified topology (referred to as 'DM+IT'), the DM using innovation estimates with predictor models obtained from true topology (referred to as 'DM+TT+IN') and identified topology (referred to as 'DM+IT+IN'), and the EBDM using innovation estimates with predictor models obtained from true topology (referred to as 'EBDM+TT+IN') and identified topology (referred to as 'EBDM+IT+IN').

The true model orders of all the modules in the network are assumed to be known. The length of impulse response for EBDM and EBLDM are chosen to be $\ell = 50$. The initial condition for the hyperparameters of EBDM and EBLDM are $\lambda_k = 0.5$ and $\beta_k = 0.5$. The initial condition for the parameters of target module (θ) are randomly chosen for EBDM and EBLDM.

To evaluate the performance of the methods, we use the standard goodness-of-Fit metric,

$$\text{Fit} = 1 - \frac{\|g_{76} - \hat{g}_{76}\|_2}{\|g_{76} - \bar{g}_{76}\|_2}$$

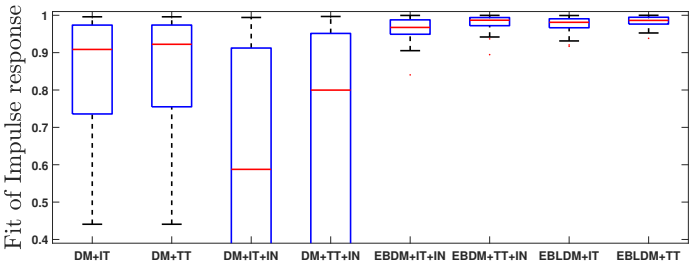


Fig. 6: Box plot of the fit of the impulse response of $\hat{G}_{76}$ obtained over 50 MC simulations for different identification methods

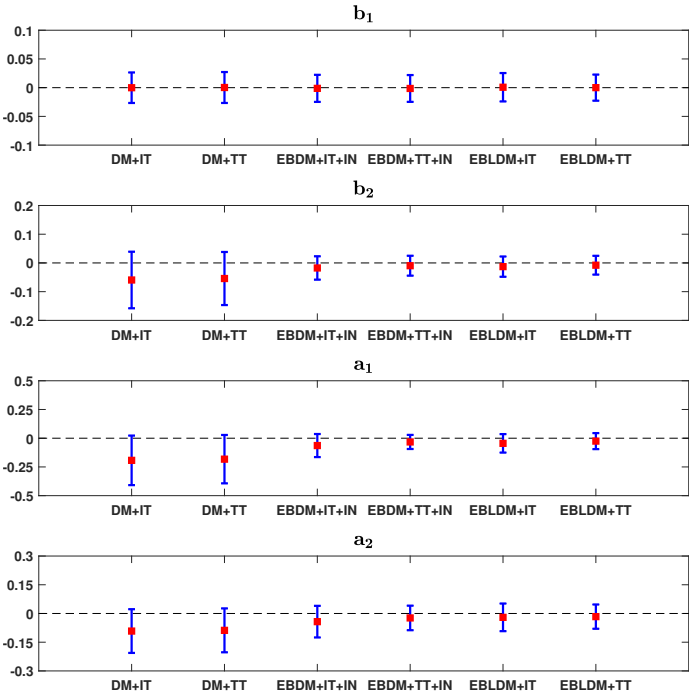


Fig. 7: Bias and standard deviation of the parameters of $\hat{G}_{76}$ obtained over 50 MC simulations for different identification methods

where, g_{76} is the true value of the impulse response of G_{76}, $\hat{g}_{76}$ is the impulse response of the estimated $\hat{G}_{76}$ and $\bar{g}_{76}$ is the sample mean of g_{76}.

The box plot showing the fit of the impulse response of G_{76} for the different methods is shown in Figure 6. It can be observed that the fit of the impulse response for the identified topology is very similar to the true topology, implying that the use of identified topology considerably preserves the performance with respect to the fit of impulse response. In the Figure 6, we also see that the EBLDM and the EBDM have significantly better fit compared to the DM. Moreover, we see that the DM using innovation signals has poor performance while the EBDM has a good performance. The poor performance of the DM can be attributed to the non-convex optimization where we obtain a local solution to the optimization. Although EBDM has a non-convex optimization, the non-parametric modelling of the modules in this method reduces the number of local minima in the optimization problem resulting in better estimation.

The bias and standard deviation of the parameters of $\hat{G}_{76}$ is shown in Figure 7. It can be observed that the use of identified topology does not affect the estimation bias. Although, the standard deviation observed in the estimates are slightly higher for identified topology compared to true topology, the estimates obtained using the EBLDM and EBDM are superior to those of DM. This reduction in variance can be credited to the regularization approach followed in both the EBLDM and EBDM. On further analysis, a clear bias is evident in the estimates obtained using the DM, which is in accordance to the result of [26] that under the effect of correlated process noise, a MISO structure of identification leads to biased estimate. Incorporating the innovation estimate into the estimation problem has a positive effect for EBDM whose performance is comparable to that of EBLDM except for a marginally higher estimation bias. Therefore, the developed algorithm has an adequate performance in identifying the local topology. Moreover, the target module estimate obtained by using the identified topology is quite similar to the estimate obtained using the true topology.

The example considered in this study is a network of moderate size. However, when the size of the network increases, the locality of the output node need not necessarily expand. Additionally, even if the locality of the output node expands, the local topology identification can run in parallel due to its MISO formulation. Moreover, the estimation steps are convex and have analytical expressions requiring no optimization. As a result, the developed algorithm will be scalable to larger networks.

VII. Discussion and Future Work

While deriving the sparsity conditions of the non-causal Wiener filter, the modules in G are not restricted to strictly proper transfer functions. This restriction is applied only during the innovation estimation and the structure selection steps. This restriction is applied because a feedthough term in the network affects the identification of the direction of causality. This phenomenon can be observed in the topology identification studies [12], [14], [39], where only an undirected network topology is obtained for networks with proper G. Identifying the target module for proper networks whose topology is unknown is therefore considered a topic of future research.

The search algorithm proposed in this study is a focused approach to identifying the topology. The efficiency of the search algorithm and the choice of AIC as the selection criteria are motivated from a practical standpoint. However, results such as consistency and variability of the identified structure can be explored further.

Moreover, in this study, we consider implementing a non-causal Wiener filter for identifying the locality. This choice is due to the ease of implementation of a non-causal Wiener filter. The sparsity condition for non-causal and causal Wiener filters are the same [14]. Therefore, following a different method of Wiener filter implementation or opting for a causal Wiener filter will not significantly affect the algorithm's performance.

VIII. Conclusion

A novel algorithm for identifying a module embedded in a dynamic network effectively and efficiently without any prior topology information has been developed. In this study, conditions for sparsity of non-causal Wiener filters in networks with correlated process noise are derived. Using these results, the locality of the output node is identified and the network size reduced. The local network and noise topology is estimated through a series of convex optimizations with parallel computation capabilities using an algorithm that reduces the number of iterations to identify a suitable structure. Based on the identified local topology, the predictor model is built using the full input case of the local direct method and the target module is estimated minimizing the mean square error while simplifying model order selection and reducing the number of nuisance parameters. From numerical simulations, the local topology estimated using the focus search algorithm is shown to have a better performance than the available greedy search algorithm. The target module estimated with the use of identified topology is shown to have comparable performance to the estimate obtained with the use of true topology.

Appendix I
Proof of Lemma 2

In (3), we model $v(t)$ as a stationary stochastic process, such that $v(t) = H(z)e(t)$, where $e(t)$ is stationary white noise vector. Therefore, (3) becomes,

$$w(t) = (\mathcal{I} - G(z))^{-1}v(t) = (\mathcal{I} - G(z))^{-1}H(z)e(t) \quad (26)$$

Therefore, all $w_j(t)$ for $j = 1, \ldots, L$, can be written in the following form.

$$w_j(t) = \sum_{k=1}^{m} \Omega_k(z)e_k(t) \quad (27)$$

Here, $\Omega \in \mathcal{F}$ since $(\mathcal{I} - G(z))^{-1}$ and $H(z)$ have no poles on the unit circle. The vector of stationary white noise $e(t) \in \mathcal{E}$, therefore $e_k(t) \in \mathcal{E}$.

Appendix II
Proof of Theorem 1

In order to prove Theorem 1, we define the following lemma. This lemma highlights the relationship between non-causal Wiener entries and inverse of the cross-spectral density matrix Φ_w in a dynamic network.

Lemma 3: Consider the vector space $\mathcal{FE}$ containing processes $w_1, \ldots, w_L$. Define $w := (w_1, \ldots, w_L)^\top$. The processes w_i and w_j, given the processes $\{w_k\}_{k\neq i,j}$ and $i \neq j$, are non-causally Wiener-uncorrelated if and only if the entries (i, j) and (j, i) of $\Phi_w^{-1}(z)$ are zero.

Proof: Without loss of generality, consider $j = L$ and define $\bar{w} := (w_1, \ldots w_{L-1})^\top$ and $\bar{v} := (v_1, \ldots v_{L-1})^\top$. Since in (3), the process noise may be correlated, we consider the following decomposition of the process noise v_L.

$$v_L = (v_L)^{\perp\bar{v}} + (v_L)^{\|\bar{v}}. \quad (28)$$

Suppose, the non-causal Wiener filter estimating w_L from $\bar{w}$ is W, then,

$$w_L = W(z)\bar{w} + (v_L)^{\perp\bar{v}} \quad (29)$$

From (5), we see that $(v_L)^{\perp\bar{v}}$ is uncorrelated to $\bar{w}$. Also, $(v_L)^{\perp\bar{v}}$ is non-zero since the dynamic network has a full rank process noise. Define $r = (\bar{w}^\top \quad (v_L)^{\perp\bar{v}})^\top$. We observe that,

$$r = \begin{pmatrix} \mathcal{I} & 0 \\ -W(z) & 1 \end{pmatrix} w, \text{ and } \quad w = \begin{pmatrix} \mathcal{I} & 0 \\ W(z) & 1 \end{pmatrix} r.$$

Following this,

$$\begin{aligned}\Phi_w^{-1} &= \begin{pmatrix} \mathcal{I} & W(z)^* \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \Phi_{\bar{w}}^{-1} & 0 \\ 0 & \Phi_{(v_L)^{\perp\bar{v}}}^{-1} \end{pmatrix} \begin{pmatrix} \mathcal{I} & 0 \\ W(z) & 1 \end{pmatrix} \\ &= \begin{pmatrix} \Phi_{\bar{w}}^{-1} + W(z)^*W(z)\Phi_{(v_L)^{\perp\bar{v}}}^{-1} & W(z)^*\Phi_{(v_L)^{\perp\bar{v}}}^{-1} \\ W(z)\Phi_{(v_L)^{\perp\bar{v}}}^{-1} & \Phi_{(v_L)^{\perp\bar{v}}}^{-1} \end{pmatrix}\end{aligned}$$

It can be seen that any process w_i is non-causally Wiener uncorrelated to w_L if the element at (i, L) and (L, i) in Φ_w^{-1} are zero, thus proving the assertion. ∎

Therefore, in order to prove Theorem 1, it is sufficient to check the sparsity of Φ_w^{-1}. To this end, consider (3) and rewrite it as follows

$$w(t) = (\mathcal{I} - G(z))^{-1}H(z)e(t). \quad (30)$$

Therefore, the cross-power spectral density of $w(t)$, Φ_w can be written as

$$\Phi_w = (\mathcal{I} - G)^{-1}\Phi_v(\mathcal{I} - G)^{-*}.$$

Thus, Φ_w^{-1} can be written as

$$\begin{aligned}\Phi_w^{-1} &= (\mathcal{I} - G)^*\Phi_v^{-1}(\mathcal{I} - G) \\ &= \Phi_v^{-1} - G^*\Phi_v^{-1} - \Phi_v^{-1}G + G^*\Phi_v^{-1}G.\end{aligned}$$

Considering the j^{th} row of Φ_w^{-1},

$$\begin{aligned}(\Phi_w^{-1})_{j*} = (\Phi_v^{-1})_{j*} - (G_{*j})^*\Phi_v^{-1} - (\Phi_v^{-1})_{j*}G + \\ (G_{*j})^*\Phi_v^{-1}G\end{aligned}$$

Considering the i^{th} column of $(\Phi_w^{-1})_{j*}$

$$\begin{aligned}(\Phi_w^{-1})_{ji} = (\Phi_v^{-1})_{ji} - (G_{*j})^*(\Phi_v^{-1})_{*i} - (\Phi_v^{-1})_{j*}G_{*i} + \\ (G_{*j})^*\Phi_v^{-1}G_{*i} \quad (31)\end{aligned}$$

The first term in (31) is zero if the process noise of nodes i and j are uncorrelated. The second and third terms in the expression are zero if no children of i have process noise correlated with j and vice versa. The last expression is zero if the nodes i and j have no common children or if the process noise of the children of the nodes i and j are uncorrelated. Therefore, if the node i is not in the $\mathcal{B}_j$, then $(\Phi_w^{-1})_{ji}$ is zero.

Appendix III
Computation of non-causal Wiener filter [33]

The Wiener filter determines the optimal projection of a signal $w_j(t)$ in the space $\mathcal{W}_j = \text{span}\{w_i(t + p) : p \in \mathbb{Z}\}_{i\neq j}$ Here we compute the Wiener filter by approximating it with a finite impulse response (FIR) filter, also known as FIR Wiener

filter. Let the order of the FIR Wiener filter be F. Here the optimal estimate $\hat{w}_j(t)$ is written as,

$$\hat{w}_j(t) = \sum_{\substack{k=1 \\ k\neq j}}^{L} \sum_{p=-F}^{F} h_{k,p} w_k(t+p) \tag{32}$$

The Wiener filtering orthogonality condition stated in (5) is used to determine the constants $h_{k,p}$ in $\hat{W}_j(z)$. According to the orthogonality condition,

$$\mathbb{E}\left[\hat{w}_j(t) w_i(t+p)\right] = \mathbb{E}\left[w_j(t) w_i(t+p)\right] \tag{33}$$

where, $i \in \{1, \cdots, j-1, j+1, \cdots, L\}, p \in \{-F, -F+1, \cdots, F-1, F\}$. Combining (32) and (33),

$$\begin{aligned}[R_{w_1 w_i}(-F-p) \cdots R_{w_1 w_i}(F-p) \cdots R_{w_L w_i}(-F-p) \cdots \\ R_{w_N w_i}(F-p)]\, h = R_{w_j w_i}(-p),\end{aligned} \tag{34}$$

where, $i \in \{1, \cdots, L\} \setminus j, p \in \{-F, \cdots, F\}$,

$$\begin{aligned} h &:= \left[h_1^\top h_2^\top \cdots h_{j-1}^\top h_{j+1}^\top \cdots h_N^\top\right]^\top, \text{and} \\ h_i^\top &:= [h_{i,-F} \cdots h_{i,-1} h_{i,0} h_{i,1} \cdots h_{i,F}] \end{aligned} \tag{35}$$

The set of equations in (34) and (35) describe $(2F+1)(L-1)$ linear equations in $(2F+1)(L-1)$ unknowns in the vector h Thus, in combined form the equations become,

$$Rh = S$$

Thus, $h = R^{-1}S$ is used to compute the coefficients of the Wiener filters. Note that the matrix R and the vector S can be computed using the data $\{w_k\}_{t=1}^{N}, k \in \{1, \cdots, L\}$.

Appendix IV
Proof of Proposition 2

Starting with the (8), we represent the network as follows:

$$\breve{H} w_{\mathcal{B}} = \breve{G} w_{\mathcal{B}} + \xi_{\mathcal{B}}, \tag{36}$$

where, $\breve{H} = \tilde{H}^{-1}$ and $\breve{G} = \tilde{H}^{-1}\tilde{G}$. Due to the presence of off-diagonal terms in $\tilde{H}$, $\breve{H}$ has off-diagonal terms and $\breve{G}$ has diagonal terms. As a result, we group the diagonal and off-diagonal terms of $\breve{H}$ and $\breve{G}$ into separate transfer function matrices as follows:

$$\begin{aligned} \breve{H} &= \breve{H}_D + \breve{H}_{ND}, \\ \breve{G} &= \breve{G}_D + \breve{G}_{ND}, \end{aligned} \tag{37}$$

where, the subscripts D and ND represent diagonal and non-diagonal respectively. Note that $\breve{H}_{ND}$ is strictly proper since $\breve{H}$ is monic. Incorporating (37) in (36) followed by some matrix manipulation results in,

$$(\breve{H}_D + \breve{G}_D) w_{\mathcal{B}} = (\breve{G}_{ND} + \breve{H}_{ND}) w_{\mathcal{B}} + \xi_{\mathcal{B}} \tag{38}$$

By substituting $Q = (\breve{H}_D + \breve{G}_D)$ and $P = (\breve{H}_{ND} + \breve{G}_{ND})$, we get (9).

Appendix V
Simulated Network

For the Monte Carlo simulations, we consider the following network, where $G_{ji} = \frac{B_{ji}}{1+F_{ji}}$, $H_{jj} = \frac{1+C_{jj}}{1+D_{jj}}$ and $H_{ji} = \frac{C_{ji}}{1+D_{ji}}$

$$\begin{bmatrix} B_{21} \\ B_{32} \\ B_{36} \\ B_{34} \\ B_{45} \\ B_{51'} \\ B_{57} \\ B_{1'9} \\ B_{89} \\ B_{18} \\ B_{73} \\ B_{96} \\ B_{79} \\ B_{61} \\ B_{76} \end{bmatrix} = \begin{bmatrix} -0.9860 & 0.1185 \\ -1.2690 & -0.1723 \\ -2.6700 & 0.5670 \\ -0.0470 & -0.4662 \\ 0.2100 & -0.1612 \\ 0.8930 & -0.2876 \\ 2.9140 & 0.4512 \\ 0.2520 & -0.0059 \\ -0.0500 & -0.0300 \\ -0.8200 & -0.3788 \\ -0.5330 & -0.4707 \\ -1.4800 & -0.0480 \\ -0.1330 & -0.3093 \\ 0.3410 & -0.6901 \\ 0.1050 & -0.3465 \end{bmatrix} \begin{bmatrix} q^{-1} \\ q^{-2} \end{bmatrix} \tag{39}$$

$$\begin{bmatrix} F_{21} \\ F_{32} \\ F_{36} \\ F_{34} \\ F_{45} \\ F_{51'} \\ F_{57} \\ F_{1'9} \\ F_{89} \\ F_{18} \\ F_{73} \\ F_{96} \\ F_{79} \\ F_{61} \\ F_{76} \end{bmatrix} = \begin{bmatrix} -0.9280 & 0.1784 \\ -0.7520 & -0.1633 \\ -1.4240 & 0.4838 \\ -0.0160 & -0.2540 \\ 0.3360 & -0.4127 \\ 0.3040 & -0.1567 \\ 0.9920 & 0.2458 \\ 0.1920 & -0.0072 \\ -0.0320 & -0.0307 \\ -0.3200 & -0.2365 \\ -0.2080 & -0.2939 \\ -0.5920 & -0.0307 \\ -0.1120 & -0.4168 \\ 0.1760 & -0.5699 \\ 0.0480 & -0.2534 \end{bmatrix} \begin{bmatrix} q^{-1} \\ q^{-2} \end{bmatrix} \tag{40}$$

$$\begin{bmatrix} C_{11} \\ C_{22} \\ C_{33} \\ C_{44} \\ C_{55} \\ C_{66} \\ C_{77} \\ C_{88} \\ C_{99} \\ C_{1'1'} \\ C_{18} \\ C_{45} \\ C_{54} \\ C_{79} \\ C_{81} \\ C_{97} \end{bmatrix} = \begin{bmatrix} -0.1200 & -0.1408 \\ 0.1900 & -0.0462 \\ 0.0400 & -0.0357 \\ 0.0100 & -0.0600 \\ -0.0500 & -0.0864 \\ -0.9300 & 0.2160 \\ 0 & -0.0025 \\ -0.1100 & -0.2420 \\ 0.8400 & 0.1440 \\ 0.2600 & -0.0560 \\ -0.1300 & -0.2714 \\ 0.5700 & 0.0540 \\ -0.1500 & -0.0496 \\ -0.0800 & -0.0425 \\ -0.4500 & 0.0464 \\ 1.0700 & 0.2860 \end{bmatrix} \begin{bmatrix} q^{-1} \\ q^{-2} \end{bmatrix} \tag{41}$$

$$\begin{bmatrix} D_{11} \\ D_{22} \\ D_{33} \\ D_{44} \\ D_{55} \\ D_{66} \\ D_{77} \\ D_{88} \\ D_{99} \\ D_{1'1'} \\ D_{18} \\ D_{45} \\ D_{54} \\ D_{79} \\ D_{81} \\ D_{97} \end{bmatrix} = \begin{bmatrix} -0.1920 & -0.3604 \\ 0.3040 & -0.1183 \\ 0.0640 & -0.0914 \\ 0.0160 & -0.1536 \\ -0.0800 & -0.2212 \\ -1.4880 & 0.5530 \\ 0 & -0.0064 \\ -0.1760 & -0.6195 \\ 1.3440 & 0.3686 \\ 0.4160 & -0.1434 \\ -0.2080 & -0.6948 \\ 0.9120 & 0.1382 \\ -0.2400 & -0.1270 \\ -0.1280 & -0.1088 \\ -0.7200 & 0.1188 \\ 1.7120 & 0.7322 \end{bmatrix} \begin{bmatrix} q^{-1} \\ q^{-2} \end{bmatrix} \tag{42}$$

REFERENCES

[1] D. Hayden, Y. Yuan, and J. Gonçalves, "Robust network reconstruction in polynomial time," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 4616–4621.

[2] A. Haber and M. Verhaegen, "Subspace identification of large-scale interconnected systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2754–2759, Oct 2014.

[3] H. H. M. Weerts, P. M. J. Van den Hof, and A. Dankers, "Identification of dynamic networks operating in the presence of algebraic loops," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec 2016, pp. 4606–4611.

[4] M. Zorzi and A. Chiuso, "Sparse plus low rank network identification: A nonparametric approach," *Automatica*, vol. 76, pp. 355 – 366, 2017.

[5] P. M. J. Van den Hof, A. G. Dankers, and H. H. M. Weerts, "From closed-loop identification to dynamic networks: Generalization of the direct method," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Dec 2017, pp. 5845–5850.

[6] P. M. J. Van den Hof, A. G. Dankers, and H. H. M. Weerts, "Identification in dynamic networks," *Computers & Chemical Engineering*, vol. 109, pp. 23 – 29, 2018.

[7] H. H. M. Weerts, M. Galrinho, G. Bottegal, H. Hjalmarsson, and P. M. J. Van den Hof, "A sequential least squares algorithm for ARMAX dynamic network identification," *IFAC-PapersOnLine*, vol. 51, no. 15, pp. 844 – 849, 2018, 18th IFAC Symposium on System Identification SYSID 2018.

[8] H. H. M. Weerts, P. M. J. Van den Hof, and A. G. Dankers, "Prediction error identification of linear dynamic networks with rank-reduced noise," *Automatica*, vol. 98, pp. 256 – 268, 2018.

[9] J. Adebayo, T. Southwick, V. Chetty, E. Yeung, Y. Yuan, J. Gonçalves, J. Grose, J. Prince, G. B. Stan, and S. Warnick, "Dynamical structure function identifiability conditions enabling signal structure reconstruction," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 4635–4641.

[10] J. Hendrickx, M. Gevers, and A. Bazanella, "Identifiability of dynamical networks with partial node measurements," *IEEE Trans. Autom. Control*, vol. 64, no. 6, pp. 2240–2253, 2019.

[11] X. Cheng, S. Shi, and P. M. J. Van den Hof, "Allocation of excitation signals for generic identifiability of dynamic networks," in *Proc. 58th IEEE Conf. on Decision and Control (CDC)*. IEEE, 2019, pp. 5507–5512.

[12] D. Materassi and G. Innocenti, "Topological identification in networks of dynamical systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1860–1871, Aug 2010.

[13] B. M. Sanandaji, T. L. Vincent, and M. B. Wakin, "Exact topology identification of large-scale interconnected dynamical systems from compressive observations," in *Proceedings of the 2011 American Control Conference*, June 2011, pp. 649–656.

[14] D. Materassi and M. V. Salapaka, "On the problem of reconstructing an unknown topology via locality properties of the wiener filter," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1765–1777, July 2012.

[15] A. Chiuso and G. Pillonetto, "A bayesian approach to sparse dynamic network identification," *Automatica*, vol. 48, no. 8, pp. 1553 – 1565, 2012.

[16] S. Nabavi and A. Chakrabortty, "Topology identification for dynamic equivalent models of large power system networks," in *2013 American Control Conference*, June 2013, pp. 1138–1143.

[17] S. Jahandari and D. Materassi, "Identification of dynamical strictly causal networks," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 4739–4744.

[18] S. Shi, G. Bottegal, and P. M. J. Van den Hof, "Bayesian topology identification of linear dynamic networks," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 2814–2819.

[19] P. M. J. Van den Hof, A. Dankers, P. Heuberger, and X. Bombois, "Identification of dynamic models in complex networks with prediction error methods—basic methods for consistent module estimates," *Automatica*, vol. 49, no. 10, pp. 2994 – 3006, 2013.

[20] A. Dankers, P. M. J. Van den Hof, X. Bombois, and P. Heuberger, "Errors-in-variables identification in dynamic networks — consistency results for an instrumental variable approach," *Automatica*, vol. 62, pp. 39 – 50, 2015.

[21] A. Dankers and P. M. J. Van den Hof, "Non-parametric identification in dynamic networks," in *2015 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 3487–3492.

[22] A. Dankers, P. M. J. Van den Hof, D. Materassi, and H. H. M. Weerts, "Conditions for handling confounding variables in dynamic networks," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3983 – 3988, 2017, 20th IFAC World Congress.

[23] M. Galrinho, N. Everitt, and H. Hjalmarsson, "Incorporating noise modeling in dynamic networks using non-parametric models," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10 568 – 10 573, 2017, 20th IFAC World Congress.

[24] J. Linder and M. Enqvist, "Identification and prediction in dynamic networks with unobservable nodes," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10 574 – 10 579, 2017, 20th IFAC World Congress.

[25] K. R. Ramaswamy, G. Bottegal, and P. M. J. Van den Hof, "Local module identification in dynamic networks using regularized kernel-based methods," in *2018 IEEE Conference on Decision and Control (CDC)*, Dec 2018, pp. 4713–4718.

[26] K. R. Ramaswamy and P. M. J. Van den Hof, "A local direct method for module identification in dynamic networks with correlated noise," 2019, provisionally accepted for publication in *IEEE Trans. Automatic Control*.

[27] L. Ljung, *System Identification : Theory for the User*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1999.

[28] A. Dankers, P. M. J. Van den Hof, X. Bombois, and P. S. C. Heuberger, "Identification of dynamic models in complex networks with prediction error methods: Predictor input selection," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 937–952, April 2016.

[29] M. Gevers, A. Bazanella, and G. Vian da Silva, "A practical method for the consistent identification of a module in a dynamical network," *IFAC-PapersOnLine*, vol. 51-15, pp. 862–867, 2018, proc. 18th IFAC Symp. System Identif. (SYSID2018).

[30] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[31] V. C. Rajagopal, K. R. Ramaswamy, and P. M. J. Van den Hof, "A regularized kernel-based method for learning a module in a dynamic network with correlated noise," in *2020 IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, accepted for publication.

[32] T. Kailath, A. Sayed, and B. Hassibi, *Linear Estimation*. New Jersey: Prentice Hall, 2000.

[33] S. Talukdar, D. Deka, B. Lundstrom, M. Chertkov, and M. V. Salapaka, "Learning exact topology of a loopy power grid from ambient dynamics," in *Proceedings of the Eighth International Conference on Future Energy Systems*, ser. e-Energy '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 222–227.

[34] A. Dankers, "Optimization method for obtaining estimates in a dynamic network," Tech. Rep., 8 2019.

[35] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations and gaussian processes—revisited," *Automatica*, vol. 48, no. 8, pp. 1525 – 1535, 2012.

[36] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 773–795, 1995.

[37] P. Zhang, "On the convergence rate of model selection criteria," *Communications in Statistics - Theory and Methods*, vol. 22, no. 10, pp. 2765–2775, 1993.

[38] D. M. Chickering, "Optimal structure identification with greedy search," *Journal of Machine Learning Research*, vol. 3, no. Nov, pp. 507–554, 2002.

[39] M. Dimovska and D. Materassi, "Granger-causality meets causal inference in graphical models: Learning networks via non-invasive observations," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 5268–5273.