Blind Identification of Sparse Dynamic Networks and Applications

Mustafa Ayazoglu   Mario Sznaier   Necmiye Ozay

Abstract— This paper considers the problem of identifying the topology of a sparsely interconnected network of dynamical systems from experimental noisy data. Specifically, we assume that the observed data was generated by an underlying, unknown graph topology where each node corresponds to a given time-series and each link to an unknown autoregressive model that maps those time series. The goal is to recover the sparsest (in the sense of having the fewest number of links) structure compatible with some *a-priori* information and capable of explaining the observed data. Contrary to related existing work, our framework allows for (unmeasurable) exogenous inputs, intended to model relatively infrequent events such as environmental or set-point changes in the underlying processes. The main result of the paper shows that both the network topology and the unknown inputs can be identified by solving a convex optimization problem, obtained by combining Group-Lasso type arguments with a re-weighted heuristics. As shown here, this combination leads to substantially sparser topologies than using either group Lasso or orthogonal decomposition based algorithms. These results are illustrated using both academic examples and several non–trivial problems drawn from multiple application domains that include finances, biology and computer vision.

I. INTRODUCTION

The problem of identifying the interconnection structure of a network of dynamical systems is ubiquitous in science and engineering, arising in fields as diverse as neuroscience, systems biology, chemistry and economics. The idea of identifying causal relationships between time series dates back to the pioneering work of Granger [1], introducing the concept of "Granger Causality", which formalized the intuitive idea that if a time series $\{x\}$ is "caused" by a second one $\{y\}$, then knowledge of the past values of $\{y\}$ should lead to better prediction of future values of $\{x\}$. The idea of Granger Causality, combined with graphical methods was exploited in [2], [3] to identify the underlying structure of a network where a set of time series are related through Vector Auto Regressive (VAR) models. More recently, [4] proposed an efficient algorithm for recovering a tree-like network structure, based on the use of a metric between nodes derived using Wiener filtering arguments.

In general, in the absence of additional *a-priori* information, the problem of recovering a network structure solely from output measurements is ill-posed, since typically, many such descriptions are possible. Under these circumstances, it is reasonable to impose additional criteria on the network

structure, in order to regularize the problem. In particular, in many situations, it is of interest to identify the *sparsest* network, in the sense of having the least number of interconnections, that explains the observed data. This scenario reflects the fact that when alternative models are possible, often the most parsimonious is the correct one. The resulting problem, network structure identification subject to additional sparsity constraints, has been the subject of intense research in the past few years, leading to a number of approaches. Motivated by the problem of identifying a sparse connectivity structure underlying brain functions, [5] proposed a Lasso type algorithm to identify a sparse network where each link corresponds to a first order VAR process, an idea extended in [6] to higher order processes. The main idea underlying these methods is to exploit the fact that penalizing the ℓ_1 norm of the vector of regression coefficients tends to produce sparse solution (indeed, in cases where the restricted isometry property holds [7], [8], this method will recover the sparsest solution). The use of an ℓ_1 type penalty has also been proposed to identify the structure of reaction [9] and genetic networks [10], where the goal, is, in both cases, to identify a sparse (static) matrix related to the network connectivity or its dynamics. The main difficulty with Lasso based approaches stems from the fact that enforcing sparsity of the entire vector of regressor coefficients does not necessarily result in a sparse network structure, since the resulting solution can consist of many links, each with a few coefficients. This difficulty can be circumvented by resorting to group Lasso type approaches [11], which seek to enforce *block sparsity* by using a combination of ℓ_1 and ℓ_2 norm constraints on the coefficients of the regressor. While this approach was shown to work well with artificial data in [12], there is no guarantee that it will indeed produce the sparsest network topology, due to its heuristic nature. Finally, a different approach was pursued in in [13], based on the use of a modified Orthogonal Least Squares algorithm, Cyclic Orthogonal Least Squares (COLS). As before, while the approach was shown to perform well in several examples, there is no guarantee that it will yield the sparsest solution, due to its greedy nature. In addition, this approach requires enforcing an *a-priori* limit on the number of links allowed to point to a single node, and such information may not be readily available, specially in cases where this number has high variability amongst nodes. It is also worth emphasizing that the approaches outlined above only consider exogenous inputs in the form of (typically gaussian) noise. However, many situations of practical interest also necessitate considering other types of inputs to take into account for instance environmental or

set-point changes in the underlying processes. An additional difficulty stems from the fact that these inputs are typically unmeasurable and must be inferred only from the actions on the observable system outputs.

To address these difficulties, in this paper we propose a convex optimization based approach to the problem of identifying sparse networks structures from observed noisy data, in the presence of unmeasurable (but infrequently changing) inputs. The proposed method is closest in spirit to that in [12], in the sense that it is also based on a group Lasso type argument. The main differences consist in the ability to handle unknown inputs, and in a reformulation of the problem, that allows for using a re-weighted iterative type algorithm. As illustrated in the paper with both synthetic data and several non–trivial real examples, the resulting algorithm substantially outperforms both group-Lasso and orthogonal least squares based ones.

II. STATEMENT OF THE PROBLEM

Next, we formalize the problem under consideration. We will consider data generated by an underlying directed graph $G = \{V, E\}$ structure, where each node V corresponds to a given time series, and the edges E connecting these nodes represent unknown linear time invariant systems, with a known bound on their order, and subject to unmeasurable, but rare exogenous disturbances. We will further assume that there are no self-loops, that is the measurements at the ith node, $x_i(t)$, do not depend on its past values, $x_i(t - k)$, leading to an overall model of the form:

$$x_i(t) = \sum_{j=1, j \neq i}^{P} \sum_{n=k}^{N} (a_{ji}(n) x_j(t - n)) + u_i(t) + \eta_i(t) \quad (1)$$

Here a_{ij} are unknown coefficients, $\eta_i(t)$, $\|\eta_i\|_\infty \leq \epsilon$ models the combined effects of noise and model uncertainty, and $u_i(t)$ represents an external, piecewise constant bias input that changes infrequently and is intended to account for relatively rare external events. An example of such a situation is a setpoint change due a change in environmental conditions (e.g. a source of nutrients being exhausted, necessitating the use of a different one). This situation will be modeled by assuming that the derivative of $u_i(t)$ is sparse. In this context, the problem of identifying the network structure can be precisely stated as, given *a priori* bounds ϵ on $\|\eta_t\|_\infty$ and n_r on the order of the regressors, find a set of coefficients a_{ij} and exogenous inputs u_i such that the resulting model explains the observed data. As mentioned early, in the absence of other priors this problem is ill posed and admits multiple solutions. Thus, in order to regularize the problem, we will seek the sparsest one, in the sense of the corresponding graph having the minimum number of edges. An example of this situation is illustrated in figure 1, showing a graph that has only 7 edges, out of a possible total of $2\binom{5}{2} = 20$ interconnections.

Rewriting (1) in matrix form yields:

$$\mathbf{x_i} = [\mathbf{X_i}, \mathbf{I}] \left[\mathbf{a_i}^t \mathbf{u_i}^t\right]^t + \mathbf{\eta_i} \quad (2)$$
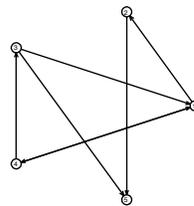

Fig. 1. Sparse Network Topology, for P=5

where

$$\begin{aligned}
\mathbf{x_i} &= [x_i(t), x_i(t-1), x_i(t-2), \cdots, x(N+1)]^t \\
\mathbf{u_i} &= [u_i(t), u_i(t-1), u_i(t-2), \cdots, u(N+1)]^t \\
\eta_i &= [\eta_i(t), \eta_i(t-1), \eta_i(t-2), \cdots, \eta(N+1)]^t \\
\mathbf{a_i} &= [a_{1i}, a_{2i}, a_{3i}, \cdots, a_{Pi}]^t
\end{aligned} \quad (3)$$

$\mathbf{I}$ denotes the identity matrix of suitable size and where

$$\mathbf{X_i} = [\mathbf{H_1}, \mathbf{H_2}, \cdots, \mathbf{H_P}] \quad (4)$$

Here $\mathbf{H_j}$ denotes the Hankel matrix formed from the measurements at the jth node, that is:

$$\mathbf{H_j} = \begin{bmatrix} x_j(t-1) & x_j(t-2) & \cdots & x_j(t-N) \\ x_j(t-2) & x_j(t-3) & \cdots & x_j(t-(N+1)) \\ \vdots & \vdots & \vdots & \vdots \\ x_j(N) & x_j(N-1) & \cdots & x_j(1) \end{bmatrix}$$

It follows that the complete network structure can be described written by a matrix equation of the form:

$$\mathbf{x} = [\mathbf{X}, \mathbf{I}] \left[\mathbf{a}^t \mathbf{u}^t\right]^t + \eta \quad (5)$$

where

$$\begin{aligned}
\mathbf{x} &= [\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}, \cdots, \mathbf{x_P}]^t \\
\mathbf{u} &= [\mathbf{u_1}, \mathbf{u_2}, \mathbf{u_3}, \cdots, \mathbf{u_P}]^t \\
\eta &= [\eta_1, \eta_2, \eta_3, \cdots, \eta_P]^t \\
\mathbf{a} &= [\mathbf{a_1}, \mathbf{a_2}, \mathbf{a_3}, \cdots, \mathbf{a_P}]^t
\end{aligned} \quad (6)$$

and

$$\mathbf{X} = \begin{bmatrix} \mathbf{X_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{X_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{X_P} \end{bmatrix}$$

Thus, in this context, the problem of interest can be formalized as finding the *block*–sparsest solution to (5)[1]

III. IDENTIFICATION ALGORITHMS

In this section we present the main results of this paper, two algorithms to search for block-sparse solutions to (5). The first algorithm is global, in the sense that it considers all the information available, and generates a complete set of links (and associated regressor vectors) and exogenous

---

[1]Note that we are not interested in sparsifying the vector $\mathbf{a}$. Rather, we want to find solutions with the minimum number of non–zero block elements $\mathbf{a}_i$.

inputs. The second algorithm is distributed, in the sense that, at a given step, only generates the edges linking a single node to the rest of the graph. While suboptimal in principle, nevertheless this algorithm is attractive for large data sets, due to its better scaling properties and robustness to perturbations.

A. Algorithm 1 (Global Optimization)

This algorithm searches for sparse solutions to (5) by solving (iteratively) the following optimization problem (suggested by the re-weighted heuristic proposed in [14])

$$\min \sum_{j=1}^{P} \sum_{i=1}^{P} W_1(j,i) \|\mathbf{a_{ji}}\|_2 + \lambda \|\mathbf{W_2 \Delta u}\|_1$$
subject to:
$$\left\| [\mathbf{X \ I}] [\mathbf{a}^t \mathbf{u}^t]^t - \mathbf{x} \right\|_\infty \le \epsilon \tag{7}$$

where $\mathbf{\Delta u}$ represents the first order difference of the input vector $\mathbf{u}$, $\mathbf{W_1}$ and $\mathbf{W_2}$ are weighting matrices, and λ is a Lagrange multiplier that plays the role of a tuning parameter between network sparsity and event sensitivity. Next, we briefly outline an heuristic to balance these values. The idea is to consider the two extreme situations where, on one case the data is explained without the need for external inputs, and in the other, the data is solely explained by the inputs. Define:

$$\mathbf{\tilde{a}} = \text{argmin} \|\mathbf{a}\|_2 \text{ subject to: } \|\mathbf{Xa} - \mathbf{x}\|_\infty \le \epsilon \tag{8}$$

or

$$\mathbf{\tilde{a}} = \text{argmin}_\mathbf{a} \|\mathbf{Xa} - \mathbf{x}\|_\infty$$

when (8) is infeasible. Next, define $\mathbf{\Delta \tilde{u}} = \mathbf{\Delta x}$, where Δ denotes first differences. In terms of these values, we propose to select λ according to:

$$\lambda = \frac{\|\mathbf{\tilde{a}}\|_2}{\|\mathbf{\tilde{u}}\|_1}$$

where the idea here is to give equal weights to the two extreme cases. The complete algorithm is outlined below

1: **procedure** REWEIGHTEDSPARSIFICATION
2:     Set $\mathbf{W_1}$ to all ones
3:     Set $\mathbf{W_2}$ to identity matrix
4:     Set δ to a small number
5:     Set ϵ to a small number
6:     Set $currval$
7:     Set $prevval$
8:     **while** $|currval - prevval| \le \epsilon$ **do**
9:         Solve (7)
10:         Set Minimizers and Minimum Value
11:         Set $prevval=currval$
12:         Set $currval$=Minimum Value
13:         Set $W_1(j,i) = 1/(\|\mathbf{a_{ji}}\|_2 + \delta)$
14:         Set Normalize $\mathbf{W_1} = \mathbf{W_1}/mean(\mathbf{W_1})$
15:         Set $\mathbf{W_2} = diag[1/(|\mathbf{\Delta u}| + \delta)]$
16:         Set Normalize $\mathbf{W_2} = \mathbf{W_2}/mean(\mathbf{W_2})$
17:     **end while**
18: **end procedure**

where $1/|\mathbf{\Delta u}|$ denotes element wise division. It is worth emphasizing that the normalization step above is required to avoid $\mathbf{W_1}$ and $\mathbf{W_2}$ becoming arbitrarily large and "poisoning" the optimization problem with some arbitrarily large numbers. For a graph with P nodes, the number of variables in the optimization problem (7) is bounded by $n = P(Pn_r + T)$, where n_r is a bound on the regressor order and T is the number of elements of each data sequence. Since the worst case computational complexity of general SDP solvers is at least $\mathcal{O}(n^8)$ [15], where n is the number of decision variables, it follows that the worst case complexity of the algorithm above is $O(P^8(Pn_r + T)^8)$.

B. Algorithm 2 (Distributed Optimization)

In this section we propose an alternative to Algorithm 1, based on a greedy, local optimization, where, at each step a node is selected, the connections to all other nodes in the graph are minimized and then the algorithm moves on to the next node. While due to its greedy nature the algorithm may fail to find the sparsest solution to the problem, it is attractive for the case of large networks, since it has lower computational complexity, In addition, the solutions obtained (due to the local nature) are more robust to changes in the network topology, since these changes affect only a subset of nodes (and hence solutions). In this case, the overall computational complexity is $O(P(Pn_r + T)^8)$, which increases at a much slower rate than that of Algorithm 1. The main idea is to solve (2) one node at a time, as outlined below[2]:

1: **procedure** LOCALSPARSIFICATION
2:     Set $i = 1$
3:     **while** $i \le P$ **do**:
4:         solve

$$\min \sum_{j=1}^{P} W_1(j,i) \|\mathbf{a_{ji}}\|_2 + \lambda \|\mathbf{W_{2,i} u_i}\|_1$$
subject to:
$$\left\| [\mathbf{X_i \ I}] [\mathbf{a_i}^t \mathbf{u_i}^t]^t - \mathbf{x_i} \right\|_\infty \le \epsilon$$

    using Procedure reWeightedSparsification
5:         Set $i = i + 1$
6:     **end while**
7: **end procedure**

IV. SYNTHETIC DATA EXPERIMENTS AND PERFORMANCE CHARACTERIZATION

In this section we apply the proposed algorithms in several examples involving artificially generated data corrupted by noise. Our goal here is to provide a comprehensive performance characterization and investigation of the effects of noise in controlled scenarios where the ground truth is available. In all cases comparisons are made both in terms of the identified structure and the (percentage) fitting error defined as:

$$PFE = \frac{\max_{i,n}(|x_i(n) - \hat{x}_i(n)|)}{\max_{i,n}(|x_i(n)|)}$$

[2]Note that this optimization can be carried out in parallel.

where x and $\hat{x}$ denote the actual data and the value predicted by the network using data from the other nodes, respectively.

Example 1: Distributed versus Centralized Identification.

This example compares the performance, both in terms of the identification results and execution times for the centralized and decentralized identification algorithms. The underlying graph and a sample trajectory are shown in figures 3(a) and 2, respectively. Finally, figure 3 compares the results obtained applying Algorithms 1 and 2 and the COLS method [13]. As shown there, both methods correctly identify the network structure, while COLS fails to do so, due to its inability to account for the external driving signal u.
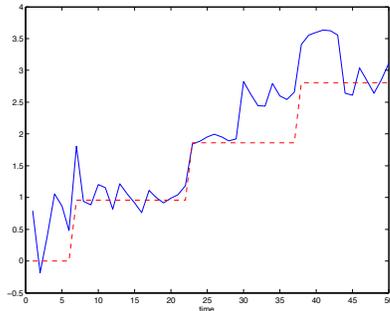

Fig. 2. An example node output (blue solid line) and its input signal (dashed red line)

(a) Ground Truth for the synthetic data experiment

(b) Structure identified using Algorithm 1 (fitting error 0%)

(c) Structure identified using Algorithm 2 (fitting error 0%)

(d) Structure identified using the COLS algorithm from [13] (fitting error 29.8%)
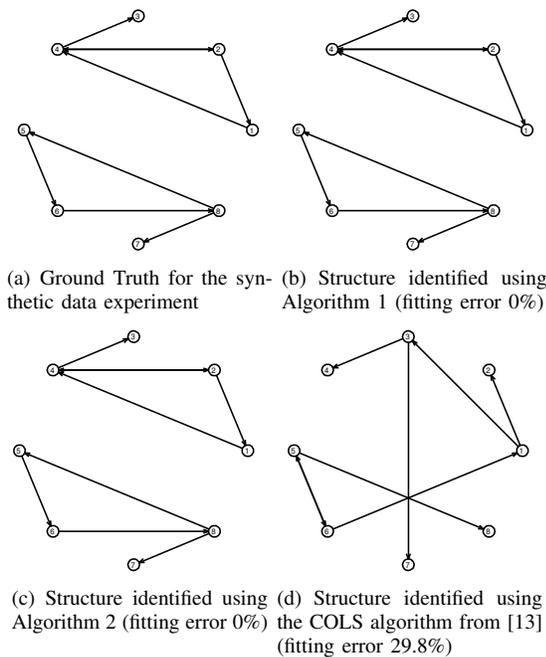
Fig. 3. Comparison of the results obtained using different algorithms for the noiseless synthetic data experiment

Finally, Table I compares the execution times of both algorithms, averaged over 25 runs. As shown there, even in a case like the present one, where the graph is small, the distributed identification method is advantageous from a computational stand point (here it is, on the average 14% faster than the centralized algorithm).

	Algorithm 1	Algorithm 2
Average Time	7.3secs	6.4secs

TABLE I

THE AVERAGE RUN TIME OF THE TWO SUGGESTED ALGORITHMS

Example 2: Effects of Data Noise. This example consists of sparse graphs with the structure shown in Fig. 3(a), where the links correspond to randomly generated second-order autoregressive models. For each realization, trajectories with 50 data points at each node were generated, starting from random initial conditions, and then corrupted with 10% additive Gaussian white noise. Figure 4 shows the network identified from the noisy trajectories using Algorithm 2, illustrating the ability of the method to recover the correct structure in the presence of noise. Similar results, omitted for space reasons, were obtained using Algorithm 1.
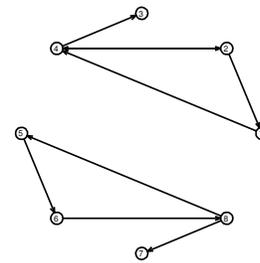

Fig. 4. Network identification from noisy data. (PFE=0.85%)

Example 3: Effects of Graph Perturbations. This example considers the case of a sparse graph perturbed by weak connections, leading to the fully connected graph shown in Figure 5. Specifically, the edges are partitioned into two sets: W (weak connections) and S (strong connections) and the connectivity level of the graph is defined by the following ratio:

$$E = \frac{max_{j \in W}(\|\mathbf{a}_j\|)}{min_{k \in S}(\|\mathbf{a}_k\|)} \tag{9}$$

Intuitively, this index measures the ratio of the strongest undesired connection to the weakest desired one. Figure 6 shows the results of applying Algorithm 2 to a graph with
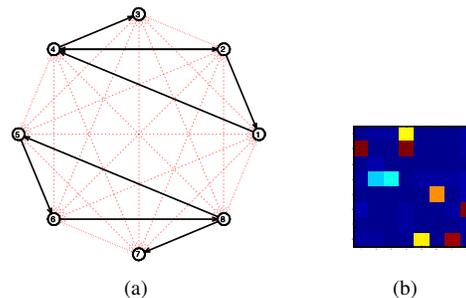

(a) (b)

Fig. 5. A sparse graph perturbed with weak connections (indicated in red). The quantitative strength of the connections is shown in the connectivity matrix, where dark blue indicates no connection and dark red a full one. The connectivity level for this example is 12%

12% sparsity level. For comparison, we also show the results of using a purely block-Lasso based algorithm, without the reweighted heuristics. As shown there, while both methods achieve a comparable value of the reconstruction error, the use of the re-weighted heuristics leads to a substantially sparser structure, effectively suppressing the weak connections.
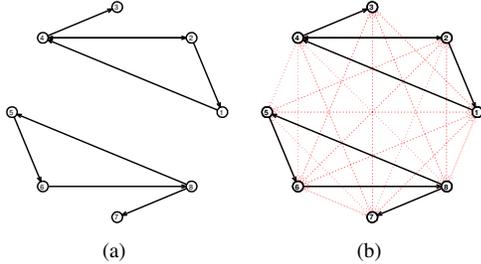

(a) (b)

Fig. 6.   Identifying a weakly perturbed graph. (a) Results obtained applying Algorithm 2 (PFE=0.38%). (b) Results obtained using a block-Lasso approach (PFE=0.38%)

V. Experiments using real data

In this section we illustrate the potential of the proposed algorithm to identify the hidden structure underlying time series drawn from different arenas, including finance, biology and computer vision.

**Example 4: Correlations and Event Detection in Financial Data.**

Name	Currency
Australian Dollar	AUD
Brazil Real	BRL
Canadian Dollar	CAD
Chinese Yuan	CNY
Danish Krone	DKK
Euro	EUR
British Pound	GBP
Hong Kong Dollar	HKD
Indian Rupee	INR
Japanese Yen	JPY
South Korean Won	KRW
Sri Lankan Rupee	LKR
Mexican Peso	MXN
Malaysian Ringgit	MYR
Norwegian Krone	NOK
New Zealand Dollar	NZD
Swedish Krona	SEK
Singapore Dollar	SGD
Thailand Baht	THB
Taiwanese Dollar	TWD
American Dollar	USD
South African Rand	ZAR

TABLE II

THE CURRENCIES USED IN THE EXPERIMENTS

This example considers a dataset similar to the one described in [13], consisting of exchange rate data for 22 different currencies, listed in Table II, for the period Jan. 01 to Dec. 31, 2007. This exchange rate data was downloaded from http://www.oanda.com/currency/historical-rates,
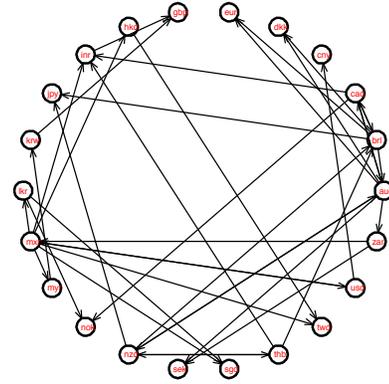
(a) Exchange rate interconnections identified using Algorithm 2
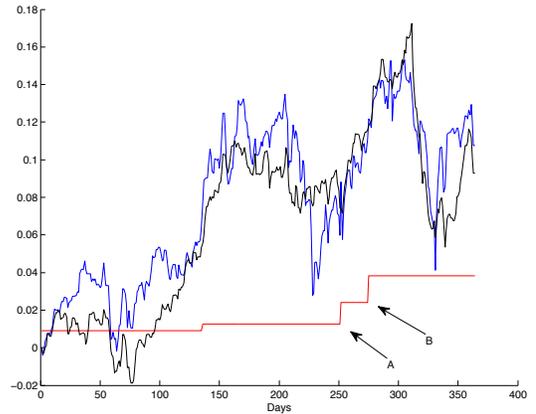
(b) Event detection using the exchange rate data

Fig. 7.   Interconnections amongst world currencies and event detection

normalized to Swiss Franks and then used to calculate a logarithmic rate of return (as in [13]), defined as:

$$r = log(\frac{V_f}{V_i}) \tag{10}$$

where V_f and V_i are the final and initial investments respectively. Figure 7(a) shows the graph structure identified applying the decentralized identification algorithm to historical values of r calculated for each of the currencies listed in Table II. While for this experiment the ground truth is unavailable, the identified interconnection structure displays features that support its validity. In particular:

1) A strong relationship was identified between the Brazilian Real and Canadian Dollar, the 8th and 9th largest world economies, respectively.
2) The strongest connection of the Australian Dollar is with the New Zealand Dollar, as it would be expected from geographical proximity.
3) One of the strongest connection of Chinese Yuan is to the United States Dollar. This is expected since the Yuan is indeed pegged to the US Dollar, rather than floating freely.

An advantage of the proposed approach (when compared for instance against that in [13]), is the additional information

provided by the identified sparse input sequence, which can be used to detect significant events. For instance, the dates of the two jumps shown in figure 7(b), Sep. 10, 2007 and Oct. 3, 2007, match the two headlines reported by the Google Insights Search Engine in 2007 for the "Brazil Real" keywords:

1) "Brazil real gains beyond 1.9 per dollar on inflows" by Reuters on Sep/13/2007
2) "Brazil Real Strengthens Beyond 1.80 Per Dollar, a 7-Year High" by Bloomberg Oct/11/2007

Example 5: A Yeast Synthetic Network. In this example we apply our method to the problem of identifying the structure underlying a synthetic network structure created in the yeast *Saccharomyces cerevisiae*, [16]. The network, shown in 8(a), is composed of five genes *(CBF1, GAL4, SWI5, GAL80, ASH1)* regulating each other through a variety of regulatory interactions, and the available data consists of measured values of galactose and glucose, once steady-state is reached after multiple perturbations. For simplicity, we only used the galactose data (16 sample points). The results of applying algorithm 2 (with and without the re-weighted heuristics) and the COLS methods are shown in Fig. 8. Table III provides a quantitative comparison of these results in terms of the criteria defined in [16], Positive Predictive Value (PPV) and Sensitivity (Se), defined as follows:

$$PPV = \frac{TP}{(TP+FP)}; \; Se = \frac{TP}{(TP+FN)} \quad (11)$$

where TP represents the true positives, FP represents the false positives and FN represents the false negatives. Note that randomly assigning edges between nodes leads to a value of $PPV = 0.7$, for all Se values, so any value higher then 0.7 is significant. As shown in the figure and table, the proposed method has substantially higher sensitivity and lower reconstruction error than COLS. The table also highlights the importance of using the re-weighted heuristics. Without it, the value of PPV drops significantly and a spurious link is added to the structure.

	PPV	Se
Algorithm 2	1	0.72
Group Lasso (No reweighted heuristic)	0.83	0.72
COLS	1	0.43

TABLE III
PPV AND Se FOR THE YEAST EXAMPLE

Example 6: Gene promoter activity in E-Coli. This example considers data involving gene promoter activity in a diauxic shift experiment using *Escherichia coli* [17]. In particular, from all the genes involved we selected 8, *(gmk,pyrG,amn,purM,wrbA,cspD,dps,cbpA)*, four of which $\{gmk, pyrG, amn, purM\}$ are nucleotide biosynthesis genes, and the rest $\{wrbA, cspD, dps, cbpA\}$ are stationary phase genes. A comparison of the identification results using different algorithms is shown in figure 9. As illustrated there, all methods are capable of segmenting the network into two disjoint components. However, Algorithm 2 leads

(a) Yeast Example Ground Truth

(b) Network identified by COLS (PFE=44.1%)

(c) Network identified using Algorithm 2 (PFE=2.7%)

(d) Network identified using Algorithm 2 without the reweighted heuristics. (PFE=2.7%)
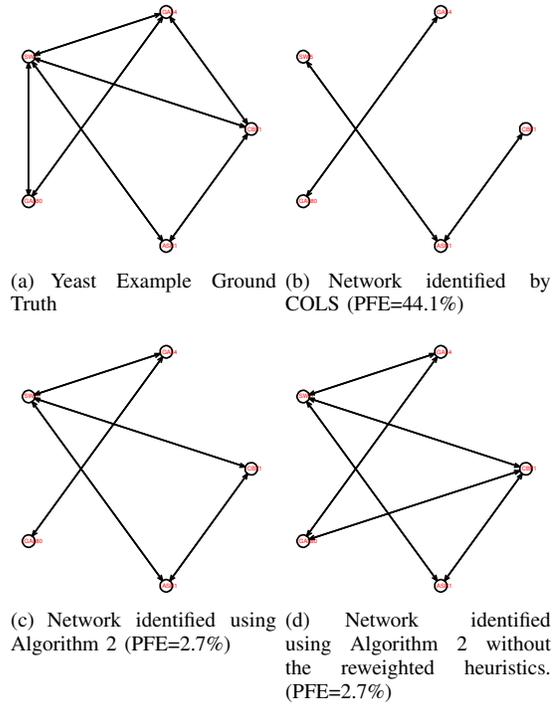
Fig. 8. Identifying the structure of an artificial yeast network

to the sparsest structure capable of explaining the available data with low reconstruction error, while COLS entails a 50% reconstruction error. It is also worth noting that, once again, the use of the re-weighted heuristics leads to a sparser network, when compared to the results of a pure block-Lasso approach.

Example 7: Identifying causality in video sequences. This example illustrates the ability of the proposed method to solve a non-trivial computer vision problem: identifying causality from video data. The data consists of 230 frames of a video clip from the Australian Open Tennis Doubles Final games. The goal here is to identify causality relationships between the different players. This can be accomplished using our formalism by modeling this scenario with a 4 node graph (each corresponding to a player), connected by autoregressive models. By restricting these models to be strictly causal, the expectation is that the network structure will strongly favor connections between members of opposite teams. Figure 10 shows sample processed frames, with the results obtained using our method and COLS super-imposed. As illustrated there, the proposed method correctly segments the teams while COLS fails to do so.

VI. CONCLUSIONS

In this paper we proposed a new method for jointly identifying the structure of an unknown network of dynamical systems and a driving input from noisy measurements. Since this problem is generically ill–posed, we propose to regularize it by searching for the sparsest structure, both in terms of links and changes in the exogenous input, capable of explaining the observed data within a given noise level. As shown in the paper, this can be accomplished by recasting the
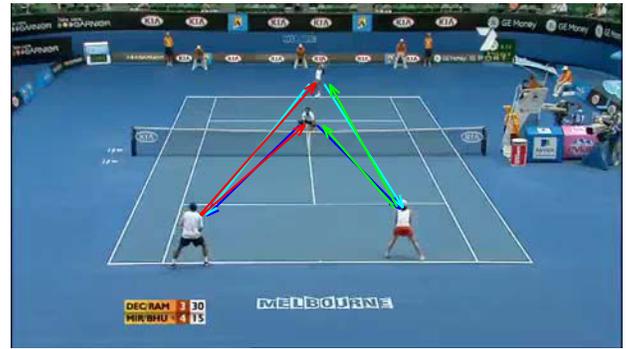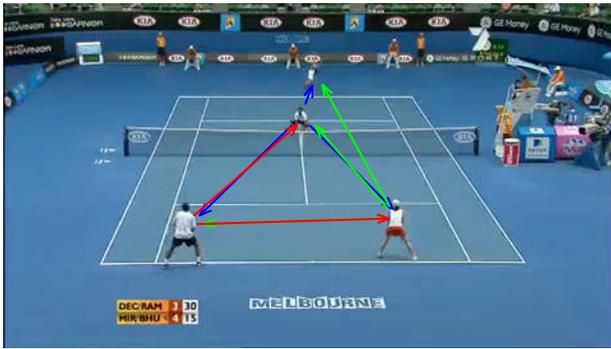
Fig. 10.   Causality from video data. Left: COLS fails to segment the teams. Right, the proposed algorithm correctly segments the teams
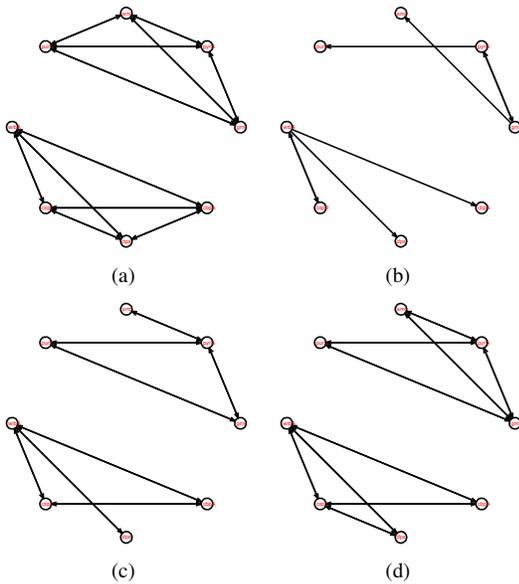

Fig. 9.   Identifying the gene promoter activation structure in an E. Coli. diauxic shift experiment. (a) Ground truth. (b) Identification via COLS (PFE=50.9%). (c)Identification using Algorithm 2 (PFE=4%). (d) Identification using Algorithm 2 without the re-weighted heuristics (PFE=4%)

problem into that of looking for block-sparse solutions to a system of linear equations, a task that we propose to solve by combining block-Lasso type arguments with an iterative re-weighted algorithm. As illustrated in the paper, the use of this re-weighted iteration leads to substantially sparser solutions than block-Lasso. In addition, the proposed algorithm also outperforms a recently proposed one, based on the use of Cyclic Orthogonal Least Squares, both in terms of the sparsity of the resulting network and the reconstruction error. These results were illustrated with both synthetic examples, highlighting the ability of the proposed method to handle noise and perturbations in the graph structure, and with several non-trivial examples drawn from applications arenas as diverse as finances, biology and computer vision.

ACKNOWLEDGMENTS

REFERENCES

[1] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, pp. 424–438l, 1969.

[2] M. Eichler, "Granger-causality and path diagrams for multivariate time series," *Journal of Econometrics*, 2002.

[3] ——, "Graphical modeling of multivariate time series with latent variables," *Preprint, Universiteit Maastricht*, 2006.

[4] D. Materassi and G. Innocenti, "Topological identification in networks of dynamical systems," *IEEE Trans. Aut. Contr.*, pp. 1860–1871, 2010.

[5] P. Valdes-Sosa, J. Sanchez-Bornot, A. Lage-Castellanos, M. Vega-Hernandez, J. Bosch-Bayard, M.-G. L., and C.-R. E., "Estimating brain functional connectivity with sparse multivariate autoregression," *Philosophical Transactions of the Royal Society*, pp. 969–981, 2005.

[6] A. Arnold, Y. Liu, and N. Abe, "Estimating brain functional connectivity with sparse multivariate autoregression," *Proc. 13th ACM Int. Conf. on Knowledge Discovery and Data Mining*, 2007, pp. 66–75.

[7] D. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inf. Th.*, 52, (1), pp. 6–18, 2006.

[8] J. Tropp, "Just relax: convex programming methods for identifying sparse signals in noise," *IEEE Trans. Inf. Th.*, 52, (3), pp. 1030–1051, 2006.

[9] A. Papachristodoulou and B. Recht, "Determining interconnections in chemical reaction networks," in *Proc. 2007 ACC*, pp. 4872 –4877.

[10] M. M. Zavlanos, A. A. Julius, and S. P. Boyd, "Identification of stable genetic networks using convex programming," in *Proc. 2008 ACC*.

[11] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables." *Journal of the Royal Statistical Society Series B*, vol. 68, no. 1, pp. 49–67, 2006.

[12] S. Haufe, G. Nolte, K. R. Muller, and N. Kramer, "Sparse causal discovery in multivariate time series," in *NIPS*, 2009.

[13] D. Materassi, G. Innocenti, and L. Giarre, "Reduced complexity models in identification of dynamical networks: Links with sparsification problems," in *48th IEEE CDC,* 2009, pp. 4796–4801.

[14] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Stanford University, 2002.

[15] T. Roh, B. Dumitrescu, and L. Vandenberghe, "Interior-point algorithms for sum-of-squares optimization of multidimensional trigonometric polynomials," in *Proc ICASSP 2007*, pp. III–905–III–908.

[16] I. Cantone, L. Marucci, F. Iorio, M. A. Ricci, V. Belcastro, M. Bansal, S. Santini, M. Di Bernardo, D. Di Bernardo, and M. P. Cosma, "A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches." *Cell*, vol. 137, no. 1, pp. 172–181, 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19327819

[17] A. Zaslaver, A. Bren, M. Ronen, S. Itzkovitz, I. Kikoin, S. Shavit, W. Liebermeister, M. G. Surette, and U. Alon, "A comprehensive library of fluorescent transcriptional reporters for Escherichia coli," *Nature Methods*, vol. 3, no. 8, pp. 623–628, Jul. 2006.