Cent. Eur. J. Phys. • 11(8) • 2013 • 1045-1055
DOI: 10.2478/s11534-013-0297-y

VERSITA

Central European Journal of **Physics**

Estimating the varying topology of discrete-time dynamical networks with noise

Research Article

Chengyi Tu*, Yuhua Cheng, Kai Chen

School of Automation Engineering,
University of Electronic Science and Technology of China,
611731 Chengdu, PR China

Received 22 April 2013; accepted 31 August 2013

Abstract: We propose an improved method to estimate the varying topology of discrete-time dynamical networks using autosynchronization. The networks considered in this paper can be weighted or unweighted and directed or undirected, and the dynamics of each node can be nonuniform. Furthermore, we suggest using a moving-average filter to suppress the influence of noise on parameter estimation. Finally, several examples are illustrated to verify the theoretical results by numerical simulation.

PACS (2008): 89.75.Fb, 89.75.Hc

Keywords: discrete-time • complex network • autosynchronization • moving average filter

1. Introduction

The study of complex dynamical networks has rapidly been attracting interest within the multidisciplinary nonlinear science community, with cell biology, ecology, computer science and meteorology being some of the many areas of investigation [1–3]. A complex dynamical network can be considered a graph where each node is a dynamical system connected to other dynamical systems through signal coupling. The connection topology plays an essential role in the cooperative behavior of networks.

To understand the cooperative dynamic behavior, and the emergent functions of a real network, one first has to identify the connection topology in terms of estimating all the elements of the coupling matrix. However, this topology estimation has not been fully investigated, and only a few methods have been developed, such as those employing Pearson's correlation [4, 5], Bayesian estimation [6], autosynchronization [7], perturbation [8], compressive-sensing [9, 10], direct reconstruction [11], linear state feedback control [12–14]. However, most of the methods make the assumption that the exact topology of the underlying network is invariable, and are only applied to estimate the topology of continuous-time dynamical networks. Meanwhile many real complex networks have varying topology, such as in the case that new nodes enter the network with time, and the available data are usually in discrete-time format in practical applications. Additionally, noise usually deteriorates the performance of parameter estimation, and leads to fluctuation of the estimates around true values in practice. However, less attention has been given to this point in previous research. Therefore, how to esti-

*E-mail: chengyitu1986@gmail.com

mate the varying topology of discrete-time dynamical networks with noise is an interesting and important subject for research. In molecular biology, the role and regulatory ability of each regulator can be obtained by sampling a time series and reconstructing the topology of a gene regulatory network [15, 16]. In neuroscience, once the brain's elements and topologies have been determined by magnetoencephalography and functional magnetic resonance imaging, they collectively form a structural network that can be explored with the tools and methods of network science [17].

The synthesis problem has motivated an approach of using synchronization as a parameter estimation method, called autosynchronization [7, 18–21]. It has recently been suggested to estimate the varying topology of discrete-time dynamical networks employing autosynchronization [22, 23], but the theory is far from reaching a state of strictness and completeness. Comparing with previous research, we systematically investigate a method that can handle varying topology and achieves faster convergence than methods described in previous papers [22, 23], and propose using a moving-average filter to suppress the amplitude of fluctuation due to noise [24].

This paper is organized as follows. Section 2 describes the problem treated in this paper. Section 3 presents preliminaries including the matrix, LaSalle's invariance principle and the moving-average filter. Section 4 presents a theoretical analysis of the varying-topology estimation of discrete-time dynamical networks. Suppressing the amplitude of fluctuation of the parameter estimation due to the presence of noise is discussed in Section 5. Subsequently, Section 6 provides examples illustrating the theoretical results, including a 3D hyperchaotic Rössler discrete-time dynamical network with and without noise. The convergence of the proposed method is demonstrated to be faster than that of the existing method [22, 23]. Larger networks, such as SW small-world and BA scale-free networks, are also given as examples. The paper is concluded in Section 7.

2. Problem statement

Consider a discrete-time dynamical network with n nodes, described by

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k)) + \sum_{j=1}^{n} a_{ij}\mathbf{h}_j\left(\mathbf{x}_j(k)\right), i = 1, 2, ..., n, \quad (1)$$

where $\mathbf{x}_i(k) = (x_{i1}(k), x_{i2}(k), ..., x_{iN}(k))^T \in \mathbf{R}^N$ is the state vector of node i, $\mathbf{f}_i(\mathbf{x}_i(k)) = (f_{i1}(\mathbf{x}_i(k)), f_{i2}(\mathbf{x}_i(k)), ..., f_{iN}(\mathbf{x}_i(k)))^T \in \mathbf{R}^N$ is the local dynamical equation of node i, and $\mathbf{h}_j\left(\mathbf{x}_j(k)\right) = \left(h_{j1}\left(\mathbf{x}_j(k)\right), h_{j2}\left(\mathbf{x}_j(k)\right), ..., h_{jN}\left(\mathbf{x}_j(k)\right)\right)^T \in \mathbf{R}^N$ is the output function of node j. The parameters a_{ij} are the elements of the uncertain coupling matrix $A = \left(a_{ij}\right)_{n\times n}$, describing the topology of the network (with $a_{ij} \neq 0$ if there is a link from node j to node i; and $a_{ij} = 0$ otherwise).

Suppose that the topology of the initial network changes at $k = k_1 \in (0, +\infty)$. There are five cases for the varying topology of discrete-time dynamical networks.

1. The nodes in the network do not change, but the weights of some edges in the network change, as illustrated from Fig. 1 (a) to (b). If new edges appear, the weights of these edges change from zero to a non-zero value; if some edges disappear, their weights change from a non-zero value to zero.

2. Nodes disappear; *i.e.*, the weights of the edges connected to the removed nodes change to zero, as illustrated from Fig. 1 (a) to (c).

3. New nodes enter the network randomly, connect with some of the existing nodes, and the edges connecting the new nodes to the existing nodes have their own weights, while the existing edges in the network are unchanged, as illustrated from Fig. 1 (a) to (d).

4. As in case 3, but the weights of the existing edges also change, as illustrated from Fig. 1 (a) to (e).

5. As in case 3, but both the weights of the existing edges and nodes of the initial network change, as illustrated from Fig. 1 (a) to (f).

Remark 1.

In cases 1 and 2, the dimensions of the coupling matrix $A = \left(a_{ij}\right)_{n\times n}$ are unchanged, while the weights of some edges change, and we can thus still use the Eq. (1) to describe them. In cases 3, 4 and 5, the dimensions of the coupling matrix increase, and the dynamical network can be rewritten as

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k)) + \sum_{j=1}^{n+n_1} \hat{a}_{ij}\left(\mathbf{h}_j\left(\mathbf{x}_j(k)\right)\right), \quad i = 1, 2, ..., n+n_1, \quad (2)$$

where $\hat{A} = \left(\hat{a}_{ij}\right) \in R^{(n+n_1)\times(n+n_1)}$ describes the topology of the new network. We can thus use Eq. (1) or (2) to describe all the cases concerning varying topology of discrete-time dynamical networks.

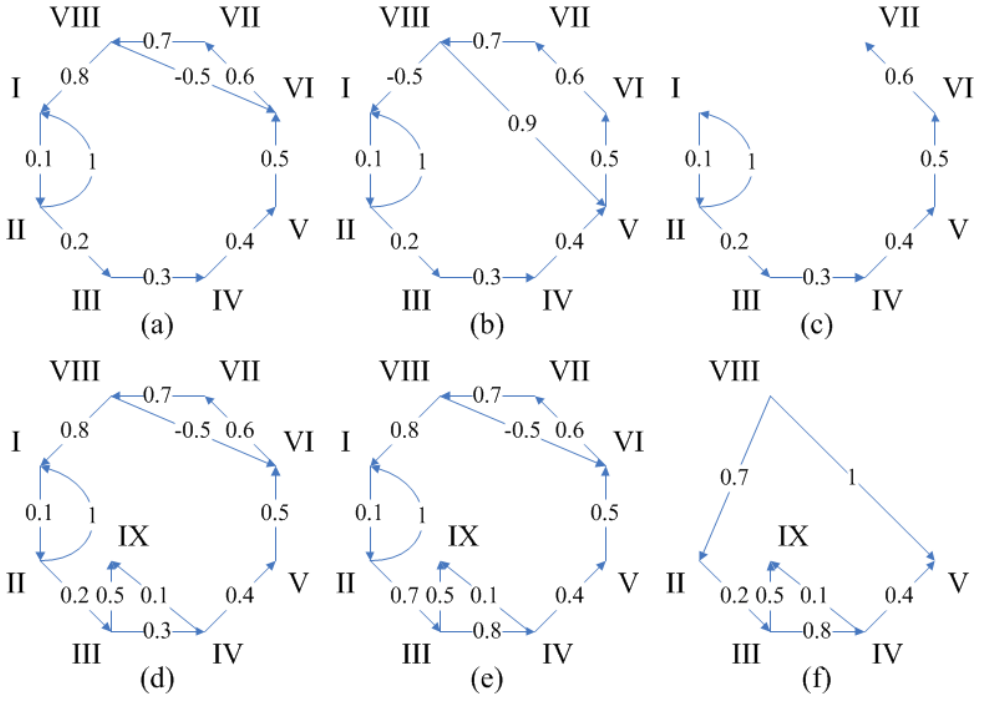

Figure 1. (a) Topology of the initial network; (b) case 1; (c) case 2; (d) case 3; (e) case 4; (f) case 5.

Remark 2.
To simplify the discussion, we assume that no nodes have coupling delays. The full description of the theory concerning coupling delays is beyond the scope of this paper and will be reported in a forthcoming paper.

3. Preliminaries

In the following section of theoretical analysis, we will use results for the matrix, LaSalle's invariance principle, and moving-average filter. Here we provide the preliminaries.

3.1. Frobenius norm and trace of the matrix

Let $\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$ denote the Frobenius norm and trA the trace of the matrix A. The following results hold:

1. $trA = trA^T = \sum_{i=1}^{n} a_{ii}, A = (a_{ij}) \in R^{n\times n}$.
2. $tr(\alpha A + \beta B) = \alpha trA + \beta trB, A, B \in R^{n\times n}, \alpha, \beta \in R$.
3. $tr(AB) = tr(BA), A \in R^{n\times m}, B \in R^{m\times n}$.
4. $\|A\|_F = \left(tr\left(AA^T\right)\right)^{\frac{1}{2}}, A = (a_{ij}) \in R^{m\times n}$.
5. $\|AB\|_F \le \|A\|_F \|B\|_F, A \in R^{m\times n}, B \in R^{n\times p}$.

3.2. LaSalle's invariance principle

Consider the difference equation

$$x(k+1) = S(x(k)), k = 0, 1, 2, \ldots, \tag{3}$$

where $S : R^N \to R^N$. Suppose $G \subset R^N$, we say V is a Lyapunov function of Eq. (3) in G, if V is continuous and $\Delta V(x(k)) = V(S(x(k))) - V(x(k)) \le 0$ for any $x(k) \in G$. Let $E = \{x : \Delta V(k) = 0, x \in \bar{G}\}$, where $\bar{G}$ is the closure of G. Let $M \subseteq E$ with $S(M) = M$, and if $M' \subseteq E$ with $S(M') = M'$, then $M' \subseteq M$. Such M is called the max-invariance set of E. Let $V^{-1}(c) = \{x : V(x) = c, x \in R^N\}$.

Theorem 3.
LaSalle's invariance principle: If (i) V is a Lyapunov function of Eq. (3) on G, and (ii) $x(k)$ is a solution of Eq. (3) bounded and in G for all $k \ge 0$, then there is a number c, such that $x(k) \to M \cap V^{-1}(c)$ when $k \to \infty$ [25].

3.3. Moving-average filter

A moving-average filter is commonly used to filter out higher-frequency components and highlight longer-term trends or cycles in time-series data. The filter takes the average of all data up until the current datum point when the data arrive in an ordered datum stream. A typical

moving-average filter of the sequence of i values $x_1, ..., x_i$ up to the current time is

$$MA_i = \frac{x_1 + ... + x_i}{i}. \tag{4}$$

4. Topology estimation theory

In the following, we address an improved method for estimating the varying topology of discrete-time dynamical networks, or more precisely, estimating the elements of the uncertain coupling matrix $A = (a_{ij})$.

Assumption 4.
In Eq. (1) or (2), we assume that functions $\mathbf{f}_i$ and $\mathbf{h}_j$ are known and bounded, and $\mathbf{x}_i(k)$ can be measured for all $i, j = 1, 2, ..., n$ or $n + n_1, k = 1, 2, ...$

Assumption 5.
$\mathbf{h}_j(\mathbf{x}_j), j = 1, 2, ..., n$ *or* $n+n_1$ *are linearly independent.*

4.1. No additional new node

In cases 1 and 2, there are no new nodes and the dimensions of the coupling matrix $A = (a_{ij})_{n\times n}$ are unchanged. Regarding Eq. (1) as a driving network, one can design a responding network:

$$\mathbf{y}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k)) + \sum_{j=1}^{n} b_{ij}(k)\,\mathbf{h}_j(\mathbf{x}_j(k)), \tag{5}$$

$$b_{ij}(k+1) = b_{ij}(k) - c\mathbf{e}_i(k+1)^T\mathbf{h}_j(\mathbf{x}_j(k)), \tag{6}$$

where $\mathbf{y}_i(k) = (y_{i1}(k), y_{i2}(k), ..., y_{iN}(k))^T \in R^N$, $b_{ij}(k) \in R$ are time series of the parameter variable, $\mathbf{e}_i(k) = \mathbf{y}_i(k) - \mathbf{x}_i(k)$, c is a variable, and $i, j = 1, 2, ..., n$.

Rewriting Eqs. (1), (5) and (6) in the matrix form, we have

$$\mathbf{X}(k+1) = \mathbf{F}(\mathbf{X}(k)) + A\mathbf{H}(\mathbf{X}(k)), \tag{7}$$

$$\mathbf{Y}(k+1) = \mathbf{F}(\mathbf{X}(k)) + \mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k)), \tag{8}$$

$$\mathbf{B}(k+1) = \mathbf{B}(k) - c\mathbf{E}(k+1)\,\mathbf{H}(\mathbf{X}(k))^T, \tag{9}$$

where $\mathbf{X}(k) = (\mathbf{x}_1(k), \mathbf{x}_2(k), ..., \mathbf{x}_n(k))^T \in R^{n\times N}$, $\mathbf{Y}(k) = (\mathbf{y}_1(k), \mathbf{y}_2(k), ..., \mathbf{y}_n(k))^T \in R^{n\times N}$, $\mathbf{E}(k) = \mathbf{Y}(k) - \mathbf{X}(k) = (\mathbf{e}_1(k), \mathbf{e}_2(k), ..., \mathbf{e}_n(k))^T \in R^{n\times N}$, $\mathbf{F}(k) = (\mathbf{f}_1(k), \mathbf{f}_2(k), ..., \mathbf{f}_n(k))^T \in R^{n\times N}$, $\mathbf{H}(k) = (\mathbf{h}_1(k), \mathbf{h}_2(k), ..., \mathbf{h}_n(k))^T \in R^{n\times N}$, $\mathbf{B}(k) = (b_{ij}(k)) \in R^{n\times n}$.

Subtracting Eq. (7) from Eq. (8), we have

$$\mathbf{E}(k+1) = (\mathbf{B}(k) - A)\,\mathbf{H}(\mathbf{X}(k)). \tag{10}$$

Furthermore, substituting Eq. (10) into Eq. (9) and then subtracting A from both sides, we have

$$\Delta\mathbf{B}(k+1) = \Delta\mathbf{B}(k)\left(\mathbf{I} - c\mathbf{H}(\mathbf{X}(k))\,\mathbf{H}(\mathbf{X}(k))^T\right), \tag{11}$$

where $\Delta\mathbf{B}(k) = \mathbf{B}(k) - A$ and $\mathbf{I}$ is an unit matrix.
We consider a Lyapunov function

$$V(k) = \sum_{i=1}^{n}\sum_{j=1}^{n}\Delta b_{ij}(k)^2 = \|\Delta\mathbf{B}(k)\|_F^2, \tag{12}$$

and use the results for the Frobenius norm and trace of the matrix in Section 3.1. As preliminaries, we know that

$$\begin{aligned}
\Delta V &= V(k+1) - V(k) = \left\|\Delta\mathbf{B}(k)\left(\mathbf{I} - c\mathbf{H}(\mathbf{X}(k))\,\mathbf{H}(\mathbf{X}(k))^T\right)\right\|_F^2 - \|\Delta\mathbf{B}(k)\|_F^2 \\
&= -2c \times tr\left[(\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k)))\,(\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k)))^T\right] \\
&+ c^2 \cdot tr\left[\left(\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\,\mathbf{H}(\mathbf{X}(k))^T\right)\left(\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\,\mathbf{H}(\mathbf{X}(k))^T\right)^T\right] \\
&= -2c\,\|\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\|_F^2 + c^2\left\|\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\,\mathbf{H}(\mathbf{X}(k))^T\right\|_F^2 \\
&\le -2c\,\|\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\|_F^2 + c^2\,\|\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\|_F^2\,\|\mathbf{H}(\mathbf{X}(k))\|_F^2 \\
&= \left(c^2\,\|\mathbf{H}(\mathbf{X}(k))\|_F^2 - 2c\right)\|\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\|_F^2
\end{aligned} \tag{13}$$

In each iterative step, $\|\Delta\mathbf{B}(k)\,\mathbf{H}(\mathbf{X}(k))\|_F^2 \ge 0$ is a fixed value. Therefore, if we want to mini-

mize $\left(c^2\left\|\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2-2c\right)\left\|\Delta\mathbf{B}\left(k\right)\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2<0$, we should minimize $-c\left(2-c\left\|\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2\right)<0$. The iterative process thus has faster convergence when

$$c\left(k\right)=\frac{1}{\left\|\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2}. \tag{14}$$

Letting $\Delta V\left(k\right)=0$, we have $\left\|\Delta\mathbf{B}\left(k\right)\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2=0$; *i.e.*,

$$\sum_{j=1}^{n}\Delta b_{ij}\left(k\right)\mathbf{h}_j\left(\mathbf{x}_j\left(k\right)\right)=0,\ i=1,2,...,n. \tag{15}$$

According to Assumption 5, $\mathbf{h}_j\left(\mathbf{x}_j\left(k\right)\right),\ j=1,2...,n$ are linearly independent. Therefore, $\Delta b_{ij}\left(k\right)=0$ from Eq. (15); *i.e.*, $\mathbf{e}_i\left(k\right)=0$ for all $i,j=1,2,...,n$. According to LaSalle's invariance principle, $\Delta b_{ij}\left(k\right)=0$ is in the max-invariance set of $\Delta V\left(k\right)=0$; i.e., $b_{ij}\left(k\right)=a_{ij},\ i,j=1,2,...,n$ is the global asymptotic attractor of the adaptive controller of Eq. (5).

Summarizing the above analysis, we obtain the following theorem for a monitor (as a program or algorithm) to be used by a computer to monitor the varying topology of discrete-time dynamical networks.

Theorem 6.

Suppose that Assumption 4 and 5 hold. In cases 1 and 2, the dimensions of the coupling matrix $\mathbf{A}=\left(a_{ij}\right)\in R^{n\times n}$ *of the discrete-time dynamical networks in Eq. (1) do not change, and the networks can be monitored by the matrix* $\mathbf{B}\left(k\right)=\left(b_{ij}\left(k\right)\right)\in R^{n\times n}$ *as follows:*

$$\begin{cases}\mathbf{X}\left(k+1\right)=\mathbf{F}\left(\mathbf{X}\left(k\right)\right)+\mathbf{A}\mathbf{H}\left(\mathbf{X}\left(k\right)\right),\\ \mathbf{Y}\left(k+1\right)=\mathbf{F}\left(\mathbf{X}\left(k\right)\right)+\mathbf{B}\left(k\right)\mathbf{H}\left(\mathbf{X}\left(k\right)\right),\\ c\left(k\right)=\dfrac{1}{\left\|\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2},\\ \mathbf{B}\left(k+1\right)=\mathbf{B}\left(k\right)-c\left(k\right)\mathbf{E}\left(k+1\right)\mathbf{H}(\mathbf{X}\left(k\right))^T.\end{cases} \tag{16}$$

Remark 7.

When employing the method of Guo and Fu [23], a larger c results in faster convergence, where $0<c<2\left(\sum_{j=1}^{n}\sum_{k=1}^{N}L_{jk}^2\right)^{-1}$, $\left|h_{jk}\left(\bullet\right)\right|\le L_{jk}, j=1,2,...,n, k=1,2,...,N$. Here we systematically investigate this relationship, and obtain $c\left(k\right)$ for convergence using our proposed method that is faster than the convergence in [23] in theory. In Section 6.2, numerical simulation is conducted to clearly demonstrate that our proposed method outperforms the method used in [23]. However, Eq. (14) is not optimal for the selection of c because ΔV has increased before the optimization.

4.2. Additional nodes

In cases 3, 4 and 5, there are new nodes and the matrix changes from $A=\left(a_{ij}\right)\in R^{n\times n}$ to $\hat{A}=\left(\hat{a}_{ij}\right)\in R^{(n+n_1)\times(n+n_1)}$ as the dimension of the coupling matrix increase. Regarding Eq. (2) as a driving network, one can design a responding network

$$\mathbf{y}_i\left(k+1\right)=\mathbf{f}_i\left(\mathbf{x}_i\left(k\right)\right)+\sum_{j=1}^{n+n_1}\hat{b}_{ij}\left(k\right)\mathbf{h}_j\left(\mathbf{x}_j\left(k\right)\right), \tag{17}$$

$$\hat{b}_{ij}\left(k+1\right)=\hat{b}_{ij}\left(k\right)-c\mathbf{e}_i(k+1)^T\mathbf{h}_j\left(\mathbf{x}_j\left(k\right)\right), \tag{18}$$

where $\mathbf{y}_i\left(k\right)=(y_{i1}\left(k\right),y_{i2}\left(k\right),...,y_{iN}\left(k\right))^T\in R^N$ and $\hat{b}_{ij}\left(k\right)\in R$ are time series of the parameter variable, $\mathbf{e}_i\left(k\right)=\mathbf{y}_i\left(k\right)-\mathbf{x}_i\left(k\right)$, c is a variable, and $i,j=1,2,...,n+n_1$.

Rewriting Eqs. (2), (17) and (18) in the matrix form, we have

$$\mathbf{X}\left(k+1\right)=\mathbf{F}\left(\mathbf{X}\left(k\right)\right)+\hat{\mathbf{A}}\mathbf{H}\left(\mathbf{X}\left(k\right)\right), \tag{19}$$

$$\mathbf{Y}\left(k+1\right)=\mathbf{F}\left(\mathbf{X}\left(k\right)\right)+\hat{\mathbf{B}}\left(k\right)\mathbf{H}\left(\mathbf{X}\left(k\right)\right), \tag{20}$$

$$\hat{\mathbf{B}}\left(k+1\right)=\hat{\mathbf{B}}\left(k\right)-c\mathbf{E}\left(k+1\right)\mathbf{H}(\mathbf{X}\left(k\right))^T, \tag{21}$$

where $\mathbf{X}\left(k\right)=\left(\mathbf{x}_1\left(k\right),\mathbf{x}_2\left(k\right),...,\mathbf{x}_{n+n_1}\left(k\right)\right)^T\in R^{(n+n_1)\times N}$, $\mathbf{Y}\left(k\right)=\left(\mathbf{y}_1\left(k\right),\mathbf{y}_2\left(k\right),...,\mathbf{y}_{n+n_1}\left(k\right)\right)^T\in R^{(n+n_1)\times N}$, $\mathbf{E}\left(k\right)=\mathbf{Y}\left(k\right)-\mathbf{X}\left(k\right)=\left(\mathbf{e}_1\left(k\right),\mathbf{e}_2\left(k\right),...,\mathbf{e}_{n+n_1}\left(k\right)\right)^T\in R^{(n+n_1)\times N}$, $\mathbf{F}\left(k\right)=\left(\mathbf{f}_1\left(k\right),\mathbf{f}_2\left(k\right),...,\mathbf{f}_{n+n_1}\left(k\right)\right)^T\in R^{(n+n_1)\times N}$, $\mathbf{H}\left(k\right)=\left(\mathbf{h}_1\left(k\right),\mathbf{h}_2\left(k\right),...,\mathbf{h}_{n+n_1}\left(k\right)\right)^T\in R^{(n+n_1)\times N}$, $\mathbf{B}\left(k\right)=\left(b_{ij}\left(k\right)\right)\in R^{(n+n_1)\times(n+n_1)}$.

Similar to the theoretical analysis in Section 4.1, we obtain the following theorem.

Theorem 8.

Suppose that Assumption 4 and 5 hold. In cases 3, 4 and 5, the dimensions of the new coupling matrix $\hat{A}=\left(\hat{a}_{ij}\right)\in R^{(n+n_1)\times(n+n_1)}$ *of the discrete-time dynamical networks in Eq. (2) increase, and the networks can be monitored by the matrix* $\hat{\mathbf{B}}\left(k\right)=\left(b_{ij}\left(k\right)\right)\in R^{(n+n_1)\times(n+n_1)}$ *in the following steps:*

$$\begin{cases}\mathbf{X}\left(k+1\right)=\mathbf{F}\left(\mathbf{X}\left(k\right)\right)+\hat{\mathbf{A}}\mathbf{H}\left(\mathbf{X}\left(k\right)\right),\\ \mathbf{Y}\left(k+1\right)=\mathbf{F}\left(\mathbf{X}\left(k\right)\right)+\hat{\mathbf{B}}\left(k\right)\mathbf{H}\left(\mathbf{X}\left(k\right)\right),\\ c\left(k\right)=\dfrac{1}{\left\|\mathbf{H}\left(\mathbf{X}\left(k\right)\right)\right\|_F^2},\\ \hat{\mathbf{B}}\left(k+1\right)=\hat{\mathbf{B}}\left(k\right)-c\left(k\right)\mathbf{E}\left(k+1\right)\mathbf{H}(\mathbf{X}\left(k\right))^T.\end{cases} \tag{22}$$

5. Estimation in the presence of noise

It is important to consider the effect of noise on synchronization, especially for applications of the proposed method. If noise can take the system out of the basin of attraction of the synchronization manifold, the parameter estimation performance will deteriorate dramatically and the proposed method will fail completely; otherwise, the noise is tolerated at that level. Even if the noise is tolerated, however, the estimated values of unknown parameters will fluctuate around the true values when the experimental outputs are disturbed by noise [24].
To suppress the fluctuation of the estimation due to noise, we suggest employing a moving-average filter:

$$\tilde{\mathbf{B}}(k) = \tilde{b}_{ij}(k) = \frac{1}{k}\sum_{l=1}^{k} b_{ij}(l), \quad i,j = 1,\ldots,n, k = 1,\ldots, \tag{23}$$

where $\mathbf{B}(k) = b_{ij}(k) \in R^{n\times n}$ is the monitor matrix of $\mathbf{A}$.

It is clear from Section 6.3 that the unknown matrix $\mathbf{A}$ can be estimated with high accuracy even in the presence of large random noise.

Remark 9.
In iterative Eq. (16) or (22), we still use the original $\mathbf{B}(k)$ or $\hat{\mathbf{B}}(k)$ in the following iteration, but $\tilde{\mathbf{B}}(k)$ in Eq. (23) is the true estimation of matrix A that we hope to obtain.

6. Numerical simulations

In this section, we perform numerical simulations to verify the theoretical results obtained in the previous section.

6.1. 3D hyperchaotic Rössler discrete-time system

Consider the 3D hyperchaotic Rössler discrete-time system [26]:

$$\begin{cases} x_1(k+1) = 3.8x_1(k)(1-x_1(k)) - 0.05(x_3(k)+0.35)(1-2x_2(k)), \\ x_2(k+1) = 3.78x_2(k)(1-x_2(k)) + 0.2x_3(k), \\ x_3(k+1) = 0.1(1-1.9x_1(k))((x_3(k)+0.35)(1-2x_2(k))-1), \end{cases} \tag{24}$$

where $|x_i(k)| \le 1, i = 1,2,3$.
We take into account the discrete-time dynamics of the networks described as the system

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k)) + \sum_{j=1}^{8} a_{ij}\mathbf{h}(\mathbf{x}_j(k)), \ i = 1,2,\ldots,8, \tag{25}$$

where $\mathbf{x}_i(k) = (x_{i1}(k), x_{i2}(k), x_{i3}(k))^T \in R^3$ (here the initial $\mathbf{x}_i(k)$ is the continuous uniform distributions with lower and upper endpoints specified by 0.89 and 0.91, respectively), $\mathbf{f}_i(\mathbf{x}_i(k))$ is chosen as the above Rössler map, and $\mathbf{h}_j(\mathbf{x}_j(k)) = (\varepsilon x_{j1}(k), \varepsilon x_{j2}(k), \varepsilon x_{j3}(k))$, $j = 1,2,\ldots 8$. The constant $\varepsilon\,(0 < \varepsilon < 1)$ represents the coupling strength(here we choose $\varepsilon = 0.005$).
Fig. 2 presents the numerical simulations of cases 1 and 2. The monitor is designed according to Theorem 6, and we set the initial values $b_{ij} = 0, i,j = 1,2,\ldots,8$. Case 1 in Fig. 2(a) shows the estimation of b_{ij} versus time for $j = 8$, illustrated from Fig. 1(a) to (b). For better visual presentation, we show $b_{i,8} + i$ versus time. Note that at k=500, $b_{1,8}, b_{5,8}$ *and* $b_{6,8}$ reach 0.8, 5, and 5.5, respectively, indicating correctly that $a_{1,8} = 0.8, a_{5,8} = 0$ *and* $a_{6,8} = -0.5$. At k=501, $a_{1,8}, a_{5,8}$ *and* $a_{6,8}$ become -0.5, 0.9, and 0, respectively. $b_{1,8}, b_{5,8}$ *and* $b_{6,8}$ then approach the correct values, again. Case 2 in Fig. 2(b) shows the topology estimation of the network from Fig. 1(a) to (c)where node 8 is removed. Initially, $b_{1,8}, b_{8,7}$ *and* $b_{6,8}$ approach $a_{1,2} = 1, a_{8,7} = 0.7$ *and* $a_{6,8} = -0.5$, respectively. At k = 501, when node 8 is removed, $b_{1,8}, b_{8,7}$ *and* $b_{6,8}$ attenuate to zero quickly.

Fig. 3 presents the numerical simulation of cases 3, 4, and 5, and the monitor is designed according to Theorem 8. Fig. 3(a) presents that at k=501, the new node 9 enters the initial network and connects with some of the existing nodes, the edges connecting the new nodes to the existing nodes have their own weights, and the existing edges in the network are unchanged, as illustrated in Fig. 1(a) to (d). We monitor $b_{9,j}, j = 1,2,\ldots,9$, and find that $b_{9,3}$ *and* $b_{9,4}$ approach to 0.5, 0.1, respectively, and the remaining nodes are correctly zero. Fig. 3(b) shows that at time k = 501, new node 9 enters the initial network and the weights of the existing edges change, as illustrated by Fig. 1(a) to (e). Initially, $a_{9,3} = 0, a_{9,5} = 0, a_{3,2} = 0.2$ *and* $a_{4,3} = 0.3$, and the values are 0.5, 0.1, 0.7, and 0.8, respectively, at k = 501. The monitor estimates these

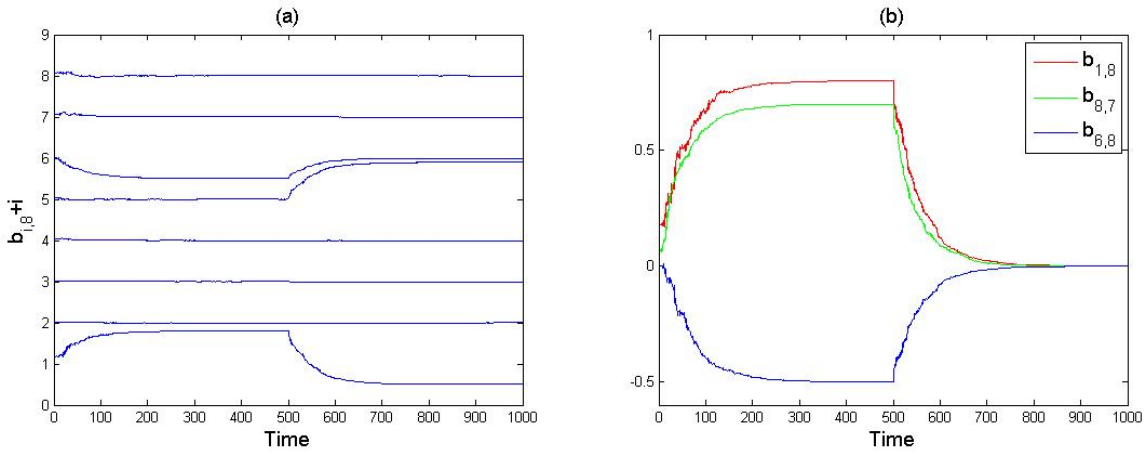

Figure 2. (a)Estimation of $b_{i,8}+i$ versus time in case 1, illustrated from Fig. 1(a) to (b); (b) the estimation of $b_{1,8}, b_{5,8}$ *and* $b_{6,8}$ versus time in case 2, illustrated from Fig. 1(a) to (c).

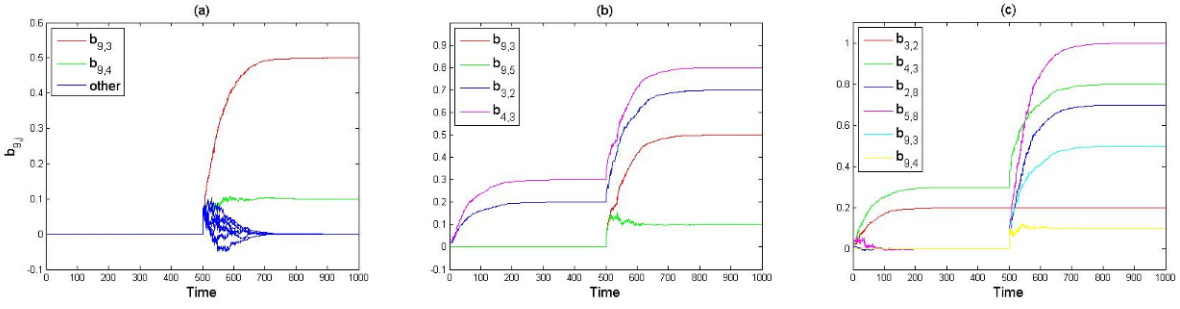

Figure 3. (a) Estimation of $b_{9,j}$ versus time in case 3, illustrated from Fig. 1(a) to (d); (b) estimation of $b_{9,3}, b_{9,5}, b_{3,2}$ *and* $b_{4,3}$ versus time in case 4, illustrated from Fig. 1(a) to (e); (c) estimation of $b_{3,2}, b_{4,3}, b_{2,8}, b_{5,8}, b_{9,3}$ *and* $b_{9,4}$ versus time in case 5, illustrated from Fig. 1(a) to (f).

values correctly. In Fig. 3(c), new node 9 enters the initial network and the weights of both the existing edges and nodes of the initial network change, as illustrated in Fig. 1(a) to (f). We monitor all the changed edges $b_{3,2}, b_{4,3}, b_{2,8}, b_{5,8}, b_{9,3}$ *and* $b_{9,4}$ of the new network. They again approach the correct values.

6.2. Comparison of existing and new methods

Guo and Fu found that for the existing method of monitoring networks, $0 < c < 2\left(\sum_{j=1}^{n}\sum_{k=1}^{N} L_{jk}^2\right)^{-1}$, $\left|h_{jk}(\bullet)\right| \leq L_{jk}, j = 1, 2, ..., n, k = 1, 2, ..., N$; a larger value of c results in faster convergence; and synchronization will occur quickly through numerical simulation when c is chosen to be the upper bound [23]. In Theorem 6, we obtain a suitable value of $c(k)$ for faster convergence by theoretical analysis. Numerical simulation is now implemented to demonstrate that the evaluation of $c(k)$ according to Eq. (14) converges more quickly than the evaluation described in [23].

To simplify the discussion, we revisit case 1 described in Fig. 2(a), and assume the initial values $x_{ij}(k) = 0.9, i = 1, 2, ..., n, j = 1, 2, 3$. We select c=1000, c=2000, c=3000, $c = \left\lfloor 2\left(\sum_{j=1}^{n}\sum_{k=1}^{N} L_{jk}^2\right)^{-1} \right\rfloor = 3333$, and $c(k) = \frac{1}{\|H(X(k))\|_F^2}$. Fig. 4(a), (b), and (c) plots $b_{1,8}, b_{5,8}$ *and* $b_{6,8}$ for five different values of c. The norm of $\left\|B(k) - A\right\|_2$ is measured to represent the level of the convergence in Fig. 4(d). It is easy to see from Fig. 4 that a larger c results in faster convergence and our new method converges more quickly than the existing method.

For better analysis of the convergence time for different values of c, we count the number of iterative steps required to reach within 0.05 of the correct value. The results are presented in Table 1. It is seen that fewer iterative steps

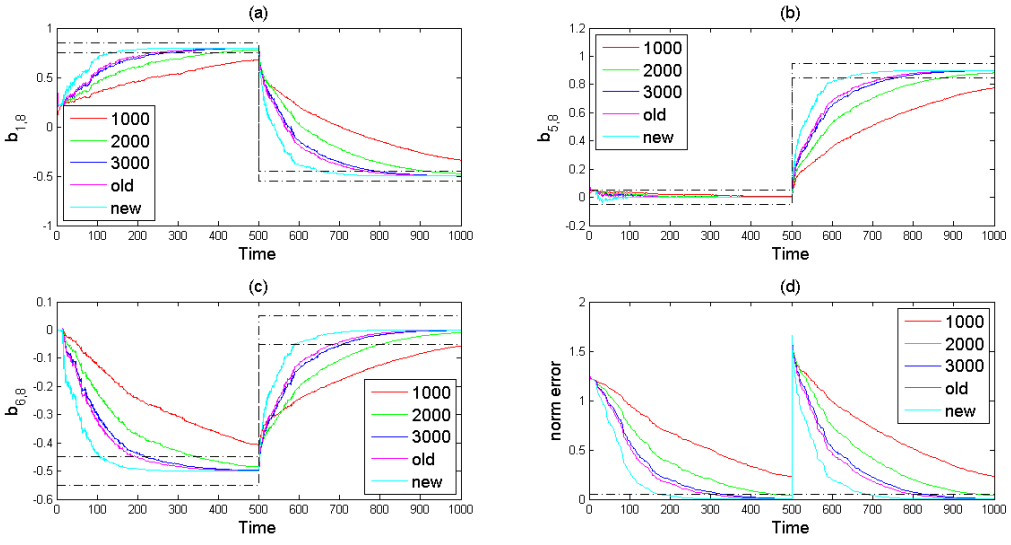

Figure 4. (a) Estimation of $b_{1,8}$ for five different values of c; (b) estimation of $b_{5,8}$ for five different values of c; (c) estimation of $b_{6,8}$ for five different values of c; (d) norm error for five values of different c. The black dashed-dotted lines show the interval within 0.05 of the correct value.

Table 1. Number of steps required to reach convergence within 0.05 of the correct value. A value of zero indicates that convergence within 0.05 of the correct value could not be achieved for the corresponding value of c. All values in the third row ($b_{5,8}$, starting at k = 1) are 1 for five different values of c because $a_{5,8} = 0$ and the initial $b_{5,8} = 0$.

	c=100	c=200	c=300	old method	new method
$b_{1,8}$, start at k=1	0	397	274	235	133
$b_{5,8}$, start at k=1	1	1	1	1	1
$b_{6,8}$, start at k=1	0	346	221	199	106
$b_{1,8}$, start at k=501	0	414	291	264	150
$b_{5,8}$, start at k=501	0	381	261	235	141
$b_{6,8}$, start at k=501	0	304	208	186	89

are needed for a larger value of c, and that our new method clearly outperforms the existing method.

6.3. Effect of noise

It is important to consider the effect of noise on synchronization, especially for applications as discussed in Section 5. As an illustrative example, we revisit the case 1 described in Fig. 2(a), but assume the output $x_{i1}(k)$, $x_{i2}(k)$ *and* $x_{i3}(k)$ are disturbed by normal random distributed noise with mean parameter zero and standard deviation parameter 0.005, respectively. We apply the moving-average filter written as Eq. (23) to suppress the fluctuation of the estimation due to noise.

To simplify the discussion, we only monitor the time series of parameter $b_{1,8}$, whose correct value is $a_{1,8} = 0.8$ at k=1 to 2000 and $a_{1,8} = -0.5$ at k=2001 to 4000. Fig. 5(a) shows the fluctuation of $b_{1,8}$ around the correct value due to noise. Fig. 5(b) shows $b_{1,8}$ after our mean-average filtering. With time, $b_{1,8}$ approaches the correct value. Fig. 5(c) shows the relative error of the estimation of $b_{1,8}$ before filtering. The relative error is disordered and follows no law. Fig. 5(d) shows the relative error in the estimation of $b_{1,8}$ after filtering. The relative error approaches zero; *i.e.*, the error becomes small with time. We clearly see the good performance of the mean-average filtering.

6.4. SW and BA networks

Let us consider larger networks, such as SW small-world network [27] and BA scale-free network [28] with 500 nodes, to illustrate the use of our method. The discrete-time dynamics of the network is described by

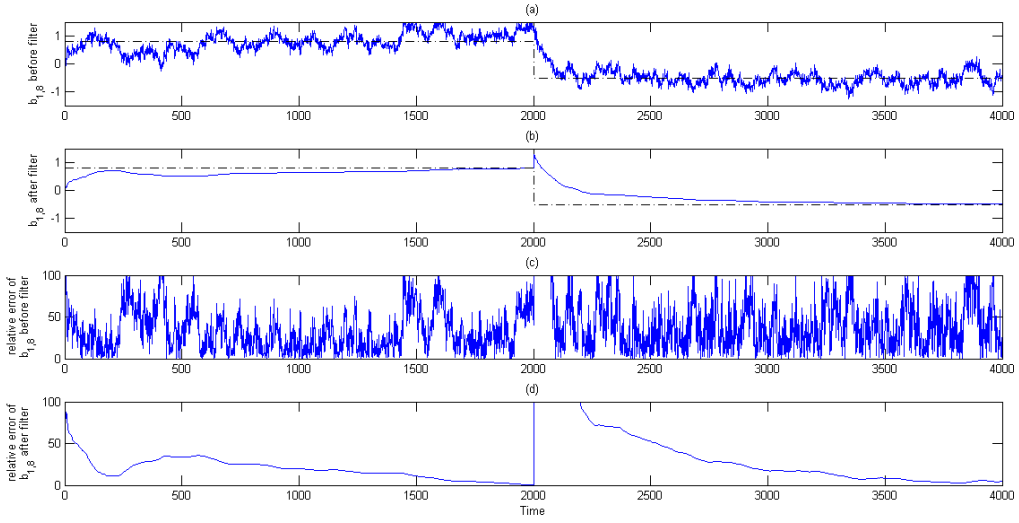

Figure 5. Estimation of $b_{1,8}$ in case 1 with noise (a) before filtering and (b) after filtering; relative error in $b_{1,8}$ (c) before filtering and (d) after filtering.

$$x_i(k+1) = f_i(x_i(k)) + \sum_{j=1}^{500} a_{ij} h_j\left(x_j(k)\right), \quad i = 1, 2, \ldots, 500, \tag{26}$$

where f is the Logistic map $f_i(x_i(k)) = 3.8x_i(k)(1 - x_i(k))$, $h_j\left(x_j(k)\right) = \varepsilon f_j\left(x_j(k)\right)$ and the constant $\varepsilon\,(0 < \varepsilon < 1)$ represents the coupling strength (here we choose $\varepsilon = 0.001$). The initial $\mathbf{x}_i(k)$ are uniform random distributions with lower and upper endpoints specified by 0 and 1, respectively.

We first create a SW network with 500 nodes, mean degree of 4, and removal probability of 0.5, and randomly remove 100 edges, randomly remove 100 nodes, and remove the 100 nodes of the largest degree at k = 5001 in separate cases. The maximum error and mean error of all elements of $|B - A|$ are shown in Fig. 6(a), (b) and (c). We then create a BA network with 500 nodes and mean degree of 4, and randomly remove 100 edges, randomly remove 100 nodes, and remove the 100 nodes of largest degree at k = 5001 in separate cases. The maximum error and mean error of all elements of $|B - A|$ are shown in Fig. 6(d), (e) and (f).

7. Conclusion and discussions

We investigated the varying-topology estimation of discrete-time dynamical networks by designing an appropriate network monitor based on autosynchronization. In particular, we systematically investigated topological estimation and finally obtained $c(k) = \frac{1}{\|\mathrm{H}(\mathrm{X}(k))\|_F^2}$ for convergence that was faster than that achieved in previous papers [22, 23]. Numerical simulation also demonstrated that c(k) for the proposed method converged more quickly than that for the existing method. Furthermore, we suggested using a moving-average filter to suppress the effect of noise in experimental outputs. We presented several illustrative examples of application, including a 3D hyperchaotic Rössler discrete-time dynamical network with and without noise. The performances of proposed and existing methods were compared in application. Furthermore, larger networks, such as SW small-world and BA scale-free networks, were presented to demonstrate the effectiveness of the proposed method. The proposed method provides a general tool for topology estimation of discrete-time dynamical networks (weighted or unweighted, directed or undirected, and possibly nonuniform dynamics of each node), so long as 4 and 5 hold. The method could be applied to gene regular networks, stock markets, brain networks, chemical oscillators, and coupled circuits, to name just a few applications. For example, we propose constructing an improved discrete dynamical system model to reconstruct the gene regulatory network and estimating the varying topology using the proposed method. This discerns not only the role of the activator or repressor for each specific regulator, but also the regulatory ability of the regulator to the transcription rate of the target gene. We are conducting experimental research with the proposed method, the results of which will be reported

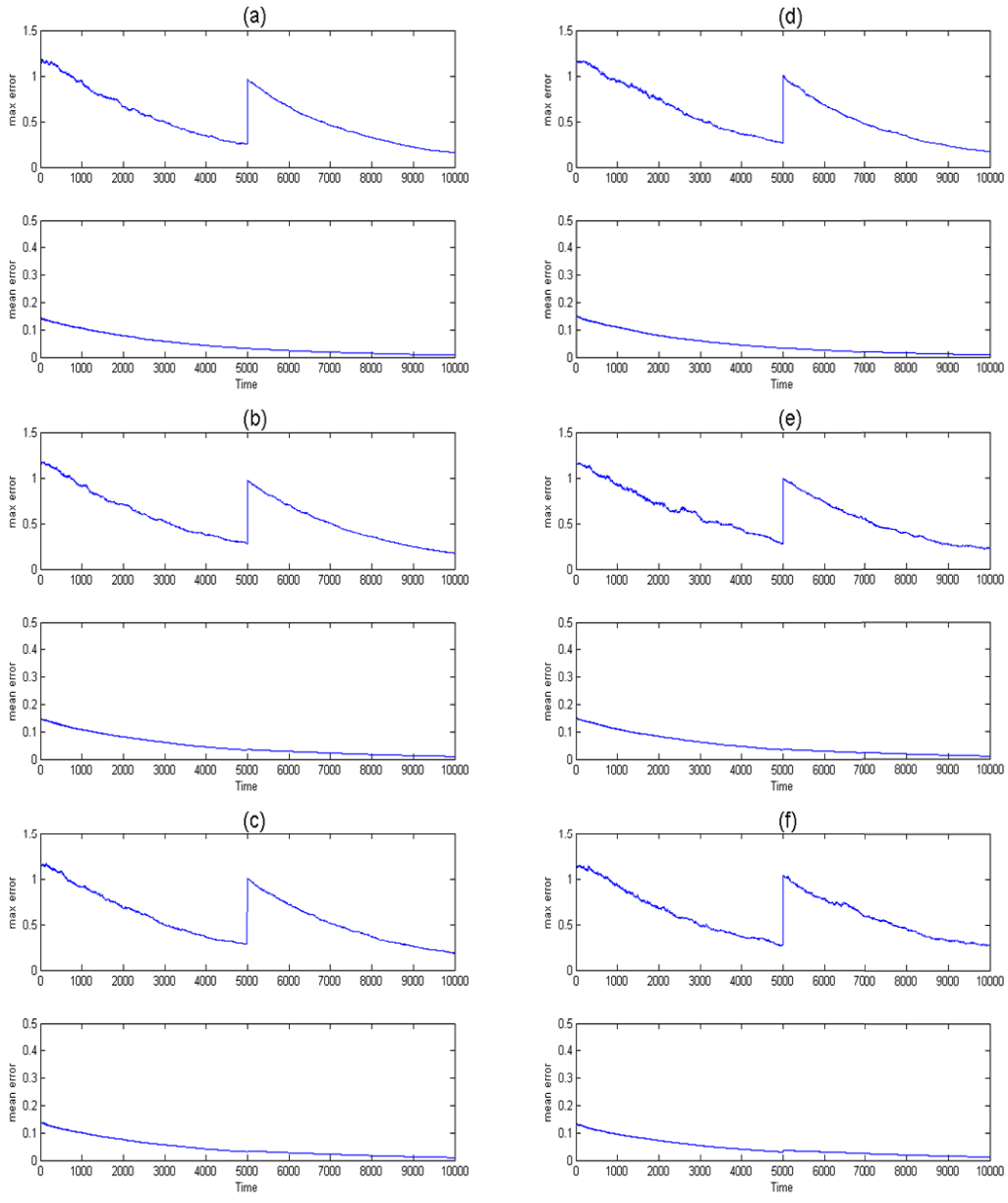

Figure 6. (a) SW network with the random removal of 100 edges; (b) SW network with the random removal of 100 nodes; (c) SW network with the removal of the 100 nodes of largest degree; (d) BA network with the random removal of 100 edges; (e) BA network with the random removal of 100 nodes; (f) BA network with the removal of the 100 nodes of largest degree.

elsewhere.

References

[1] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D. Hwang, Phys. Rep. 424, 175 (2006)
[2] L.d.F. Costa et al., Adv. Phys. 60, 329 (2011)
[3] M. E. J. Newman, Networks: An Introduction (Oxford University Press Inc., New York United States, 2010)
[4] D. S. Bassett, A. Meyer–Lindenberg, S. Achard, T. Duke, E. Bullmore, P. Natl. Acad. Sci. USA 103, 19518 (2006)
[5] V. M. Eguíluz, D. R. Chialvo, G. A. Cecchi, M. Baliki, A. V. Apkarian, Phys. Rev. Lett. 94, 018102 (2005)
[6] S. Pajevic, D. Plenz, PLoS Comp. Biol. 5, e1000271 (2009)
[7] D. Yu, M. Righero, L. Kocarev, Phys. Rev. Lett. 97,

188701 (2006)
[8] M. Timme, Phys. Rev. Lett. 98, 224101 (2007)
[9] W.-X. Wang, R. Yang, Y.-C. Lai, V. Kovanis, C. Grebogi, Phys. Rev. Lett. 106, 154101 (2011)
[10] W.-X. Wang, R. Yang, Y.-C. Lai, V. Kovanis, M. A. F. Harrison, Europhys. Lett. 94, 48006 (2011)
[11] S. G. Shandilya, M. Timme, New J. Phys. 13, 013004 (2011)
[12] D. Yu, U. Parlitz, Europhys. Lett. 81, 48007 (2008)
[13] D. Yu, Automatica 46, 2035 (2010)
[14] D. Yu, U. Parlitz, Phys. Rev. E. 82, 026108 (2010)
[15] H. C. Chen et al., Bioinformatics 20, 1914 (2004)
[16] F. M. Lopes et al., J Comput. Biol. 18, 1353 (2011)
[17] O. Sporns, Networks of the Brain (The MIT Press, Cambridge, Massachusetts, London, England, 2011)
[18] A. Pikovsky, M. Rosenblum, J. Kurths, Synchronization A universal concept in nonlinear sciences (Cambridge University Press, New York, United States of America, 2001)
[19] G. V. Osipov, J. Kurths, C. Zhou, Synchronization in Oscillatory Networks (Springer, Berlin, 2007)
[20] A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, C. Zhou, Phys. Rep. 469, 93 (2008)
[21] U. Parlitz, Phys. Rev. Lett. 76, 1232 (1996)
[22] G. Shu-Juan, F. Xin-Chu, Commun. Theor. Phys. 54, 181 (2010)
[23] S.-J. Guo, X.-C. Fu, J. Phys. A–Math. Theor. 43, 295101 (2010)
[24] D. Yu, Phys. Rev. E. 77, 066221 (2008)
[25] J. P. LaSalle, The Stability of Dynamical Systems (Hamilton Press, Berlin, New Jersey, USA, 1976)
[26] Z. Yan, Chaos. 16 (2006)
[27] D. J. Watts, S. H. Strogatz, Nature 393, 440 (1998)
[28] A.-L. Barabasi, R. Albert, Science 286, 509 (1999)