



**Cite this article:** Su R-Q, Wang W-X, Wang X, Lai Y-C. 2016 Data-based reconstruction of complex geospatial networks, nodal positioning and detection of hidden nodes. *R. Soc. open sci.* **3**: 150577. <http://dx.doi.org/10.1098/rsos.150577>

Received: 26 October 2015

Accepted: 26 November 2015

**Subject Category:**

Physics

**Subject Areas:**

complexity/mathematical modelling

**Keywords:**

time-series analysis, network reconstruction, geospatial network, compressive sensing

**Author for correspondence:**

Ying-Cheng Lai

e-mail: [ying-cheng.lai@asu.edu](mailto:ying-cheng.lai@asu.edu)

# Data-based reconstruction of complex geospatial networks, nodal positioning and detection of hidden nodes

Ri-Qi Su<sup>1</sup>, Wen-Xu Wang<sup>3</sup>, Xiao Wang<sup>2</sup> and Ying-Cheng Lai<sup>1</sup>

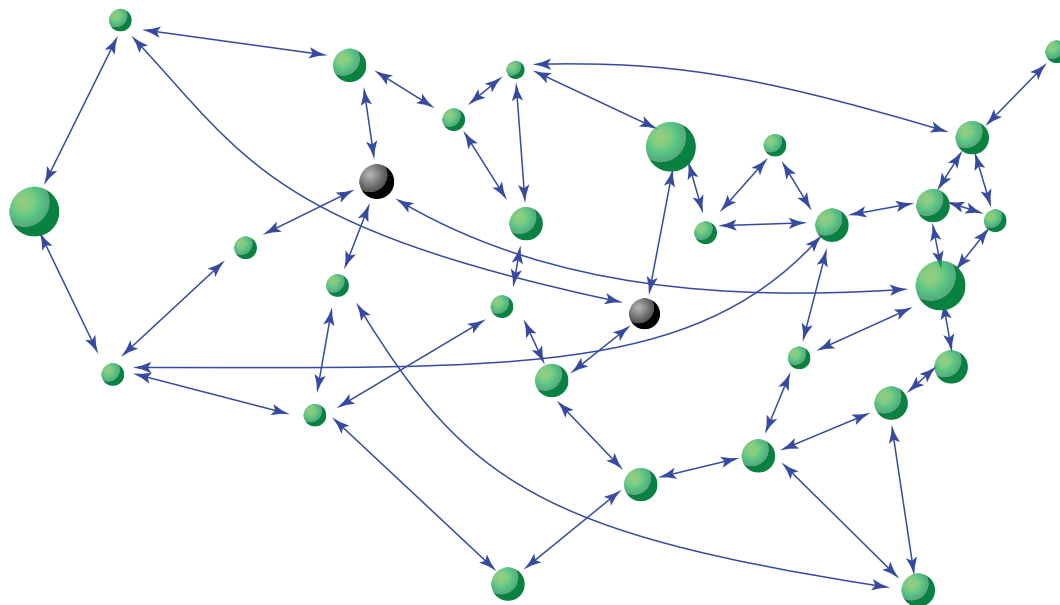
<sup>1</sup>School of Electrical, Computer, and Energy Engineering, and <sup>2</sup>School of Biological and Health Systems Engineering, Arizona State University, Tempe, AZ 85287, USA

<sup>3</sup>Department of Systems Science, School of Management and Center for Complexity Research, Beijing Normal University, Beijing 100875, People's Republic of China

Given a complex geospatial network with nodes distributed in a two-dimensional region of physical space, can the locations of the nodes be determined and their connection patterns be uncovered based solely on data? We consider the realistic situation where time series/signals can be collected from a single location. A key challenge is that the signals collected are necessarily time delayed, due to the varying physical distances from the nodes to the data collection centre. To meet this challenge, we develop a compressive-sensing-based approach enabling reconstruction of the full topology of the underlying geospatial network and more importantly, accurate estimate of the time delays. A standard triangularization algorithm can then be employed to find the physical locations of the nodes in the network. We further demonstrate successful detection of a hidden node (or a hidden source or threat), from which no signal can be obtained, through accurate detection of all its neighbouring nodes. As a geospatial network has the feature that a node tends to connect with geophysically nearby nodes, the localized region that contains the hidden node can be identified.

## 1. Introduction

Complex geospatial networks with components distributed in real geophysical space are an important part of the modern infrastructure. Examples include large-scale sensor networks and



**Figure 1.** A schematic illustration of a complex geospatial network. The connection topology, the positions of the nodes in the physical space and nodal dynamical equations are unknown *a priori*, but only time series from the nodes can be collected at a single node in the network (e.g. a data-collecting centre). The challenges are to reconstruct the dynamical network, to locate the precise position of each node and to detect hidden nodes, all based solely on time series with inhomogeneous time delays. The green circles denote ‘normal’ nodes and the dark circles indicate hidden nodes.

various subnetworks embedded in the Internet. For such a network, often the set of active nodes depends on time: the network can be regarded as static only on a relatively short time scale. For example, in response to a certain breaking news event, a communication network within the Internet may emerge, but the network will dissolve itself after the event and its impacts fade away. The connection topologies of such networks are usually unknown but in certain applications, it is desirable to uncover the network topology and to *determine the physical locations of various nodes* in the network. Suppose time series or signals can be collected from the nodes. Due to the distributed physical locations of the nodes, the signals are time delayed. Is it possible to uncover the network topology, estimate the time delays embedded in the signals from different nodes, and then determine their physical locations? Another issue is the existence of hidden nodes that cannot be directly accessed. Can the existence of a hidden node be ascertained and its location be determined?

Figure 1 illustrates a geospatial network. Assume there is a monitoring centre that collects data from nodes at various locations, but their precise geospatial coordinates are unknown. The normal nodes are coloured in green. There are also hidden nodes that can potentially be the sources of threats (e.g. those represented by dark circles). The challenging task is to determine the network topology and to locate the hidden nodes, based on time series or data only.

Data-based reconstruction of complex networks in general is deemed to be an important problem and has attracted continuous interest, where the goal is to uncover the full topology of the network based on simultaneously measured time series [1–26]. For instance, methodology was proposed to estimate the network topology controlled by feedback or delayed feedback [8,17]. Network connectivity can be reconstructed from the collective dynamical trajectories using response dynamics [7,16]. The approach of random phase resetting was introduced to reconstruct the details of the network structure [14]. For neuronal systems, there was a statistical method to track the structural changes [20,22]. While many of these previous works required complete or partial information about the intrinsic dynamics of the nodes and their coupling functions, completely data-driven and model-free methods exist. For example, the global climate network was reconstructed using the mutual information method, enabling global energy and information flow in the network to be studied [25]. The sampling bias of DNA sequences in viruses from different regions can be used to reveal the geospatial topologies of influenza networks [26]. Network structure can also be obtained by calculating the causal influences among the time series based on the Granger causality method [5,23], the transfer entropy method [21] or the method of inner composition alignment [15]. However, such causality-based methods are unable to reveal information

about the nodal dynamical equations. In addition, there were regression-based methods [27] for systems identification based on, for example, the least-squares approximation through the Kronecker-product representation [28], which would require large amounts of data.

In this paper, we develop a compressive-sensing-based framework [29–32] as a potential solution to estimating time delay and detecting hidden nodes in complex geospatial networks. To be able to fully reconstruct dynamical systems using only time series data is possible because the dynamics of many natural and man-made systems are determined by smooth enough functions that can be approximated by finite series expansions. The task then becomes that of estimating the coefficients in the series representation of the vector field governing the system dynamics. In general, the series can contain high-order terms, and the total number of coefficients to be estimated can be quite large. This is in general a challenging problem. However, if most coefficients are zero (or negligible), the vector constituting all the coefficients will be sparse. The problem of sparse vector estimation can then be solved by the paradigm of compressive sensing [29,30,32–34] that reconstructs a sparse signal from limited observations. Since the observation requirements can be relaxed considerably as compared to those associated with conventional signal reconstruction schemes, compressive sensing has evolved into a powerful technique to reconstruct sparse signal from small amounts of observations that are much less than those required in conventional approaches.

Compressive sensing has recently been introduced to the field of network reconstruction for discrete time and continuous time nodal dynamics [35,36], for evolutionary game dynamics [37], for detecting hidden nodes [38,39], for predicting and controlling synchronization dynamics [40], and for reconstructing spreading dynamics based on binary data [41]. Differing from these existing works, the focus of the present work is on estimating time delays of the dynamics at various nodes using time series collected from a single location. While there were previous methods of finding time delays in complex dynamical systems, for example, based on synchronization [42], Bayesian estimation [43] and correlation between noisy signals [19], our compressive-sensing-based method provides an alternative approach that has the advantages of generality, high efficiency, low data requirement and applicability to large networks. We demonstrate that our method can yield estimates of the nodal time delays with reasonable accuracy. After the time delays are obtained, the actual geospatial locations of various nodes can be determined by using, for example, a standard triangular localization method [44]. We expect these results to be useful for applications such as locating sensors in wireless networks and identifying/detecting/anticipating potential geospatial threats [45], an area of importance and broad interest.

## 2. Results

Compressive sensing aims to solve the following convex optimization problem:

$$\min \|\mathbf{a}\|_1 \quad \text{subject to } \mathbf{G} \cdot \mathbf{a} = \mathbf{X}, \quad (2.1)$$

where  $\mathbf{a}$  is a sparse vector to be determined,  $\mathbf{G}$  is a (known) random projection matrix,  $\mathbf{X}$  is a measurement vector that can be constructed from the available data, and  $\|\mathbf{a}\|_1 = \sum_{i=1}^N |\mathbf{a}_i|$  is the  $L_1$  norm of vector  $\mathbf{a}$ . Compressive sensing is a paradigm of high-fidelity signal reconstruction using only sparse data [29,30,32–34], which was originally developed to solve the problem of transmitting massive datasets, such as those collected from large-scale sensor arrays. In particular, because of the high dimensionality, direct transmission of such datasets would require broad bandwidth. However, there are common situations where the datasets are sparse. For example, say a dataset of  $N$  points is represented by an  $N \times 1$  vector  $\mathbf{a}$ , where  $N$  is a large integer. Since  $\mathbf{a}$  is sparse, most of its entries are zero and only a small number of  $k$  entries are non-zero, where  $k \ll N$ . One can use a Gaussian random matrix  $\mathbf{G}$  of dimension  $M \times N$  to obtain an  $M \times 1$  vector  $\mathbf{X}$ :  $\mathbf{X} = \mathbf{G} \cdot \mathbf{a}$ , where  $M \sim k$ . Because the dimension of  $\mathbf{X}$  is much smaller than that of the original vector  $\mathbf{a}$ , transmitting  $\mathbf{X}$  would require much smaller bandwidth, provided that  $\mathbf{a}$  can be reconstructed at the receiver end of the communication channel.

For our problem of reconstructing complex geospatial networks with time delay, the task is to formulate the problem into the standard compressive sensing form of equation (2.1). This can indeed be done, for example, for oscillator networks with weighted coupling and inhomogeneous time delays [46,47]. After obtaining the time delays, a standard triangular localization method [44] can be employed to locate a large portion of nodes in the network, given that the locations of a small subset of nodes are known. A hidden node can also be detected (see Methods).

## 2.1. Reconstruction of geospatial networks based on compressive sensing

To be concrete, we present results for continuous-time oscillator networks with time-delayed couplings [46–48], where for every link, the amount of delay is proportional to the physical distance of this link. The oscillator network models have been widely used in studying neuron activity [47] and ecology [48]. In general, such a system is described by

$$\dot{\mathbf{x}}_i = \mathbf{F}_i[\mathbf{x}_i(t)] + \sum_{j=1, j \neq i}^N \mathbf{W}_{ij}[\mathbf{x}_j(t - \tau_{ij}) - \mathbf{x}_i(t)], \quad (2.2)$$

for  $i = 1, \dots, N$ , where  $\mathbf{x}_i \in \mathbb{R}^m$  is the  $m$ -dimensional state variable of node  $i$  and  $\mathbf{F}_i[\mathbf{x}_i(t)]$  is the vector field for its isolated nonlinear nodal dynamics. Consider a link  $l_{ij}$  connecting nodes  $i$  and  $j$ , and the interaction weight is given by the  $m \times m$  weight matrix  $\mathbf{W}_{ij} \in \mathbb{R}^{m \times m}$  with its element  $w_{ij}^{p,q}$  representing the coupling from the  $q$ th component of node  $j$  to the  $p$ th component of node  $i$ . For simplicity, we assume only one component of  $w_{ij}^{p,q}$  is non-zero and denote it as  $w_{ij}$ . The associated time delay is denoted as  $\tau_{ij}$ . For a modern geospatial network, the speed of signal propagation is quite high in a proper medium (e.g. optical fibre). The time delay can thus be assumed to be small and we can use the Taylor expansion to express the delay coupling terms in the networked dynamical system to the first order, e.g.  $\mathbf{x}_i(t - \tau_{ij}) \approx \mathbf{x}_i(t) - \tau_{ij}\dot{\mathbf{x}}_i$ , where  $\dot{\mathbf{x}}_i$  is the time derivative. When the coupling function between any pair of nodes is linear, in a suitable mathematical basis constructed from time-series data,  $\tilde{\mathbf{g}}^{(\nu)}[\mathbf{x}_i(t)]$ , the coupling and time-delayed terms, together with the nodal dynamical equations, can be expanded into series, and our goal is to estimate all the expansion coefficients. Using our formulation of the compressive sensing framework (see Methods), equation (2.2) can then be written in the following compact form:

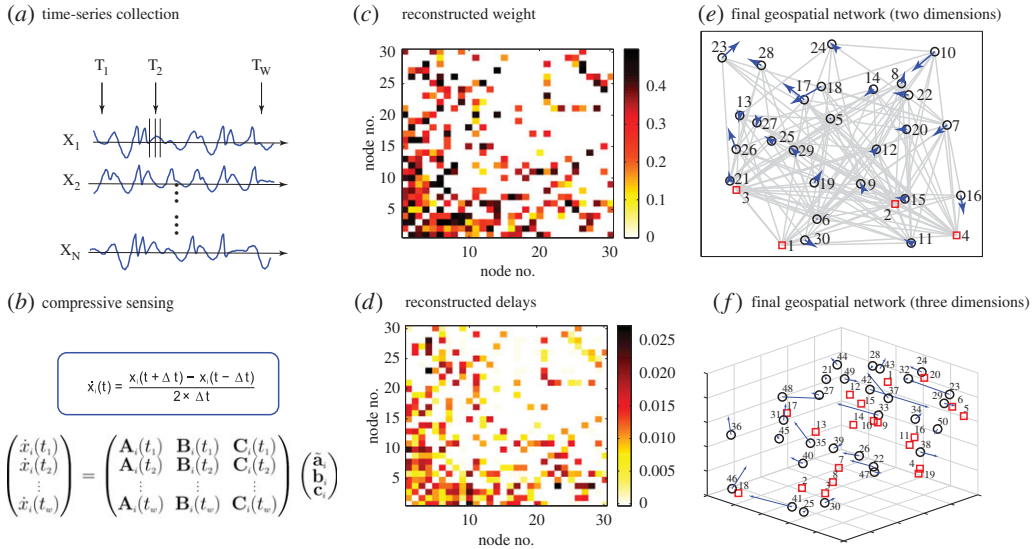
$$\dot{\mathbf{x}}_i(t) = \sum_{\nu} \tilde{\alpha}^{(\nu)} \cdot \tilde{\mathbf{g}}^{(\nu)}[\mathbf{x}_i(t)] + \sum_{j=1, j \neq i}^N [\mathbf{b}_{ij}\mathbf{x}_j(t) + \mathbf{c}_{ij}\dot{\mathbf{x}}_j(t)], \quad (2.3)$$

which is a set of linear equations for data collected at different time  $t$ . The coefficients associated with the nodal dynamical equations, the network link-weights and time delays are contained in the coefficients  $\{\tilde{\alpha}^{(\nu)}\}$ ,  $\{\mathbf{b}_{ij}\}$  and  $\{\mathbf{c}_{ij}\}$ , respectively. The amount of data required depends on the system size and the order of the series expansions, which can be small as compared with the dimension of the coefficient vectors for a properly chosen mathematical base.

After obtaining the time delays, we proceed to determining the actual positions of all nodes. If time series are collected simultaneously from all nodes at the data-collecting node, the estimated coupling delay  $\tau_{ij}$  associated with the link  $l_{ij}$  is proportional to the physical distance  $d_{ij} = d_{ji}$  of the link. However, in reality, strictly synchronous data collection is not possible. For example, if the signals are collected at a location  $s$  outside the network with varying time delays  $\tau_{si}$ , the estimated delays associated with various links in the network are no longer proportional to the actual distances. As we explain in Methods, the varying delays due to asynchronous data collection can be cancelled and the distances can still be estimated as  $d_{ij} = (c/2)(\tau_{ij} + \tau_{ji})$ , where  $\tau_{ij}$  is the signal delay associated with node  $j$  from the reconstruction of node  $i$ , vice versa for  $\tau_{ji}$ , and  $c$  is the signal propagation speed.

When the mutual distances between nodes have been estimated, we can determine the actual locations of the nodes, for example, by using the standard triangular localization algorithm [44]. This method requires that the positions of  $N_B$  reference nodes be known, the so-called beacon nodes. Starting from the beacon nodes, the triangulation algorithm makes use of the distances to these reference nodes to calculate the Cartesian coordinates of the detected nodes. The beacon node set can then be expanded with the newly located nodes. Nodes that are connected to the new beacon set, each with more than three links in two-dimensional space, can be located. The process continues until the locations of all nodes have been determined, or no new nodes can be located (see Methods). The choice of the proper initial beacon set to fully reconstruct the network depends on the network topology. For example, for a scale-free network, we can choose nodes that were firstly added during the process of network generation as the initial beacon set, thereby guaranteeing that all nodes can be located using the procedure. An empirical rule is then to designate the largest degree nodes as the initial beacon node set.

Our numerical experiments are set up as follows. We assume all nodes are distributed in a two-dimensional square, or a three-dimensional cube of unit length. The network topology can be either scale-free [49] or random [50], and the network size can be varied. For proof of principle, we consider coupled nonlinear oscillator networks by placing, at each node, a nonlinear oscillator,



**Figure 2.** Illustration of our method to reconstruct a complex geospatial network from time series. (a) For any node, time series of its dynamical variables are collected at  $w$  different instants of time. (b) The corresponding derivatives are approximated using the standard first-order Gaussian method, which are needed in constructing the compressive sensing equations. (c,d) An example of link weights and time delays obtained from the reconstructed coefficient vectors, respectively. (e) Given the positions of four beacon nodes (marked as red rectangles), the locations of the remaining nodes (marked as black circles) are determined using the standard triangularization method. The blue arrows indicate the estimation errors, which point from the actual to the estimated positions. The various coupling terms are illustrated using grey lines. There are in total 30 nodes in the network, connecting with each other via the scale-free topology. The average outgoing degree is five. The amount of data used is  $R_m = 0.5$ . (f) Following the same procedure, a geospatial network of 50 nodes which are distributed in a three-dimensional cube of unit length is reconstructed. There are 20 beacon nodes and the normalized amount of data used is  $R_m = 0.55$ . For clarity, the links between these nodes are not shown.

e.g. the Rössler oscillator, mathematically described by the following set of three first-order differential equations:  $[\dot{x}, \dot{y}, \dot{z}] = [-y - z, x + 0.2y, 0.2x + z(x - 0.2)]$ . The coupling weights are asymmetric and uniformly distributed in the interval  $[0.1, 0.5]$ . We assign a small threshold to the estimated weight as  $w_0 = 0.05$  (somewhat arbitrary), where if the estimated weight is larger (smaller) than  $w_0$ , the corresponding link is regarded as existent (non-existent). We have  $d_{ij} = c \cdot \tau_{ij} = c \cdot \tau_{ji}$  and choose  $c$  to be 100 (arbitrarily). To be specific, we choose linear coupling functions and assume that, for a pair of connected nodes, the interaction occurs between the  $z$ -variable of one node and the  $x$ -variable of another. The time series used to reconstruct the whole network system are acquired by integrating the coupled delayed differential equation system [51] with step size  $5 \times 10^{-5}$ . The vector fields of the nodal dynamics are expanded into a power series of order  $l_x + l_y + l_z \leq 3$ . The derivatives required for the compressive sensing formulation are approximated from time series by the standard first-order Gaussian method. To quantify the data requirement, we define  $R_m$  as the ratio of the number of data points used to the total number of unknown coefficients to be estimated. The beacon nodes are chosen to be those having the largest degrees in the network, and their positions are assumed to be known.

Figure 2 summarizes the major steps required for reconstructing a complex geospatial network using compressive sensing. For illustrative purpose, we first use a network of  $N = 30$  nodes that are connected with each other in a scale-free manner and are randomly distributed in a two-dimensional square. Oscillatory time series are collected from each node, from which compressive sensing equations can be obtained, as shown in figure 2a,b. The reconstructed coefficients for the nodal dynamical equations, as explained in Methods, contain the coupling weights  $B_{ij} = w_{ij}$  and the delay terms  $C_{ij} = -w_{ij} \times \tau_{ij}$ . The links with reconstructed weights larger than the threshold  $w_0$  are regarded as actual (existent) links, for which the time delays  $\tau_{ij}$  can be estimated as  $\tau_{ij} = -C_{ij}/w_{ij}$ . Repeating this procedure for all nodes, we can determine the weighted adjacency matrix (that defines the network topology) and the time delay matrix. The estimated adjacency matrix and the time delays are displayed in figure 2c,d, respectively, which match well with those of the actual network. We note that the reconstructed time delays are symmetric with respect to the link directions, as shown in figure 2d, which is correct as they depend only on the corresponding physical distances. With the estimated time delays, we choose the four largest degree nodes, nodes 1 to 4, as the beacon nodes, so that the locations of all remaining nodes can be determined.



The fully reconstructed geospatial network is shown in figure 2e, where the red rectangles indicate the locations of the beacon nodes. The black circles denote the actual locations of the remaining nodes and the heads of the blue arrows indicate their estimated positions (shorter arrows mean higher estimation accuracy). The amount of data used is relatively small:  $R_m = 0.5$ .

We further test our method in three dimensions. Using the same conditions for all nodes that are randomly distributed in a three-dimensional cube of unit length, we perform reconstruction on a scale-free network with 50 nodes, as shown in figure 2f. We follow the same method to estimate the connection weights and time delays for connected pairs, and then use triangulation localization to locate all nodes in the three-dimensional space. We employ 20 beacon nodes and use  $R_m = 0.55$ , because a higher dimensional system usually requires more reference nodes and larger data amount.

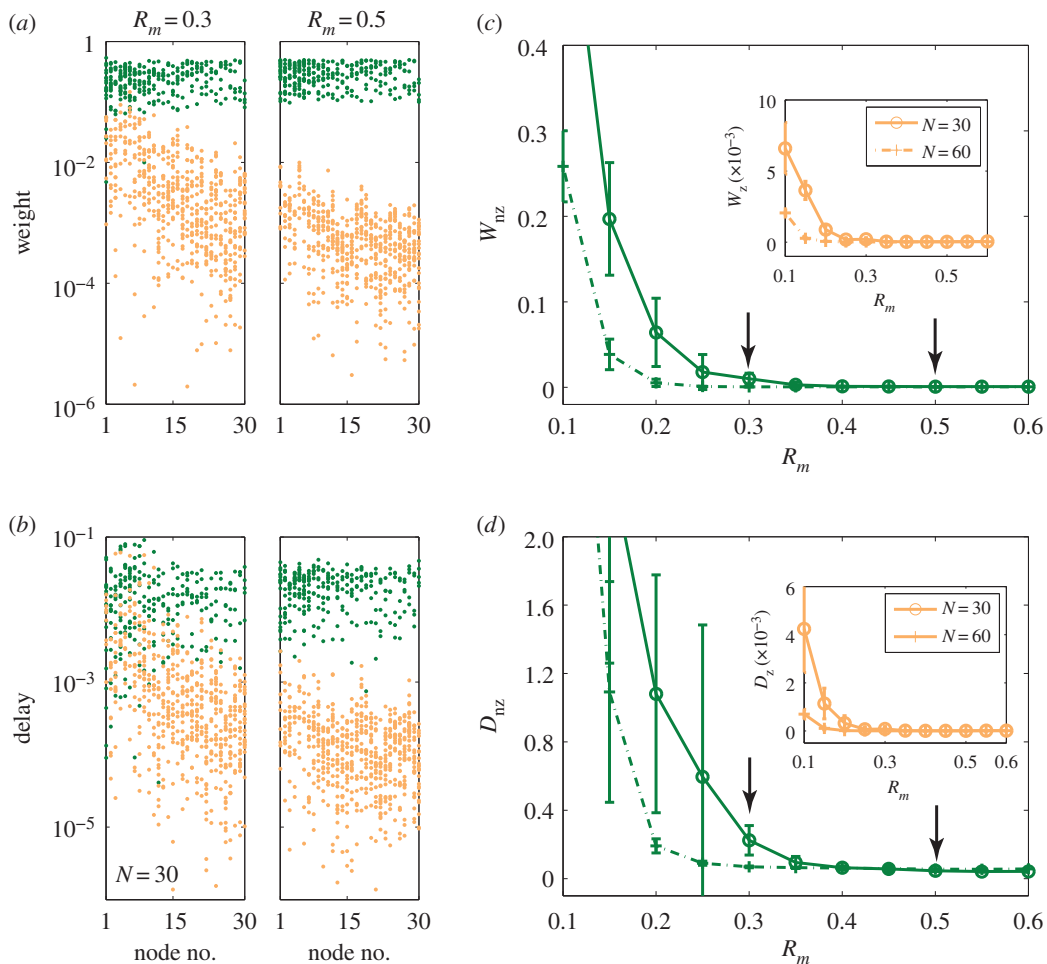
## 2.2. Performance analysis with respect to weight and time delay estimates

The performance of our compressive-sensing-based approach to reconstructing geospatial networks can be assessed by calculating the errors in the estimated weights and time delays. We define two types of errors: those associated with non-zero terms (existing links, denoted as  $W_{nz}$  and  $D_{nz}$  for weight and time delay, respectively), and those associated with zero terms (non-existing links,  $W_z$  and  $D_z$ ). In particular,  $W_{nz}$  is the error between the estimated and the true weight for an existent link, normalized by the latter, while  $W_z$  is the average absolute error associated with the original zero terms in the coefficients. Similar meanings hold for  $D_{nz}$  and  $D_z$ .

We first study the estimation errors with respect to varying data amount,  $R_m$ . Representative results are shown in figure 3 for  $R_m = 0.3$  and  $R_m = 0.5$ , where figure 3a,b are for errors in the weight and time delay estimation, respectively. For small data amount (left column), the gaps between the weights for existent and non-existent links are not well defined, especially for nodes of large degrees. As  $R_m$  is increased, the two kinds of weights can be unequivocally distinguished, making possible identification of the existent links (right column). Ensemble-averaged errors in the estimated weights and time delays versus  $R_m$  are shown in figure 3c,d, respectively. Note that, in figure 3d, the terms associated with non-zero coefficients  $C_{ij}$  are adjusted by the corresponding weights  $B_{ij}$  to compensate the actual coupling delays and the absolute errors associated with the zero coefficients, as shown in the inset, the averages of the corresponding absolute values of the  $c_{ij}$  terms. A general observation is that the various errors decrease rapidly as the data amount is increased, a prominent feature of compressive sensing.

In our mathematical formulation of the compressive-sensing-based method, the terms containing the time delays are expanded to first order only. The methodology, as it stands now, thus applies to systems with small time delays. To determine an upper bound of the time delay, below which the whole system including various time delays can be reconstructed faithfully, it is necessary to assess the dependence of the estimation errors on the amount of the time delay. Figure 4a,b show, for the special case of uniform time delay, errors  $W_{nz}$  and  $D_{nz}$  versus  $\tau$ , respectively. We see that the weight errors increase monotonically with  $\tau$ , especially for  $\tau > 10^{-2}$ . However, the time delay errors reach minimum for  $\tau \approx 10^{-2}$  and begin to increase as  $\tau$  is increased further. For relatively large time delays, the first-order Taylor expansion becomes less accurate, leading to large errors in the weight and time delay estimation. For small delays, the error in  $D_{nz}$  is due to the finite step size used in integrating the delay differential equations.

How does the performance depend on the network size and other characteristics such as the link density? Figure 5 shows, for random networks of varying size  $N$  and average connection probability  $P$ , the errors  $W_{nz}$  and  $D_{nz}$ . Specifically, for each pair of nodes in the network, their connection probability is given by  $p_{ij} \sim p_0/d_{ij}^2$ , where  $d_{ij}$  is the distance between them, and  $p_0$  is a normalization constant used to fix the average connection probability as  $P$ . In this type of ‘normalized’ networks, nodes have a larger tendency to connect to the nearby nodes, as in a real geospatial network. In figure 5a,b, the network size varies from  $N = 30$  to  $N = 100$ , while the connection density remains fixed at  $P = 0.04$ . The errors are illustrated using different colours. When the data amount  $R_m$  is increased, the errors decrease rapidly and approach a small constant value when  $R_m$  exceeds a certain critical value. We find that optimal reconstruction performance can be achieved for smaller values of  $R_m$  for networks of larger size than those of smaller size, indicating that accurate reconstruction of a larger network requires relatively smaller ratio of measurements to the number of unknown coefficients, although the absolute data amounts are larger than those for smaller networks. This is so because, as  $N$  is increased, the density of non-zero terms in the dynamical equations and coupling functions decreases for fixed connection density.

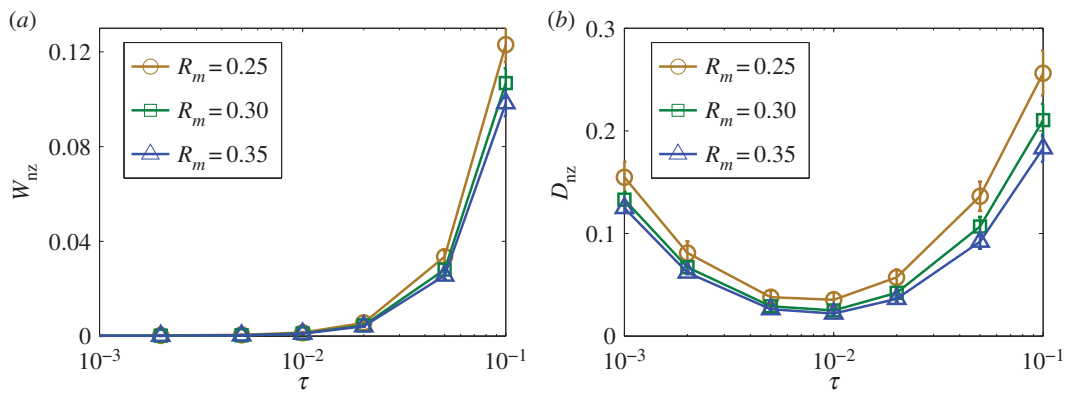


**Figure 3.** Error analysis of network reconstruction and delay time estimation. (a) Predicted incoming coupling strengths for all nodes for  $R_m = 0.3$  and  $R_m = 0.5$  on a logarithmic scale. The green and orange dots represent the weights of existent and non-existent links, respectively. (b) Predicted coefficients for the non-zero delay coupling terms  $C_{ij}$ , marked as green dots, in comparison with the estimated values for zero terms (marked as orange dots). (c) Errors in the estimates of the coupling weights associated with existent and non-existent links,  $W_{nz}$  and  $W_z$  (inset), respectively, versus the data amount  $R_m$ . (d) Errors in the estimated time delays, where the terms associated with non-zero delay coefficients  $C_{ij}$  are normalized by the corresponding weights  $B_{ij}$ . The coefficients associated with zero terms (without normalization) are shown in the inset. The green and orange curves represent results obtained from networks of size  $N = 30$  and  $N = 60$ , respectively. All errors are obtained by averaging over 20 independent network realizations. The black arrows indicate the  $R_m$  value used to calculate the results in panels (a) and (b).

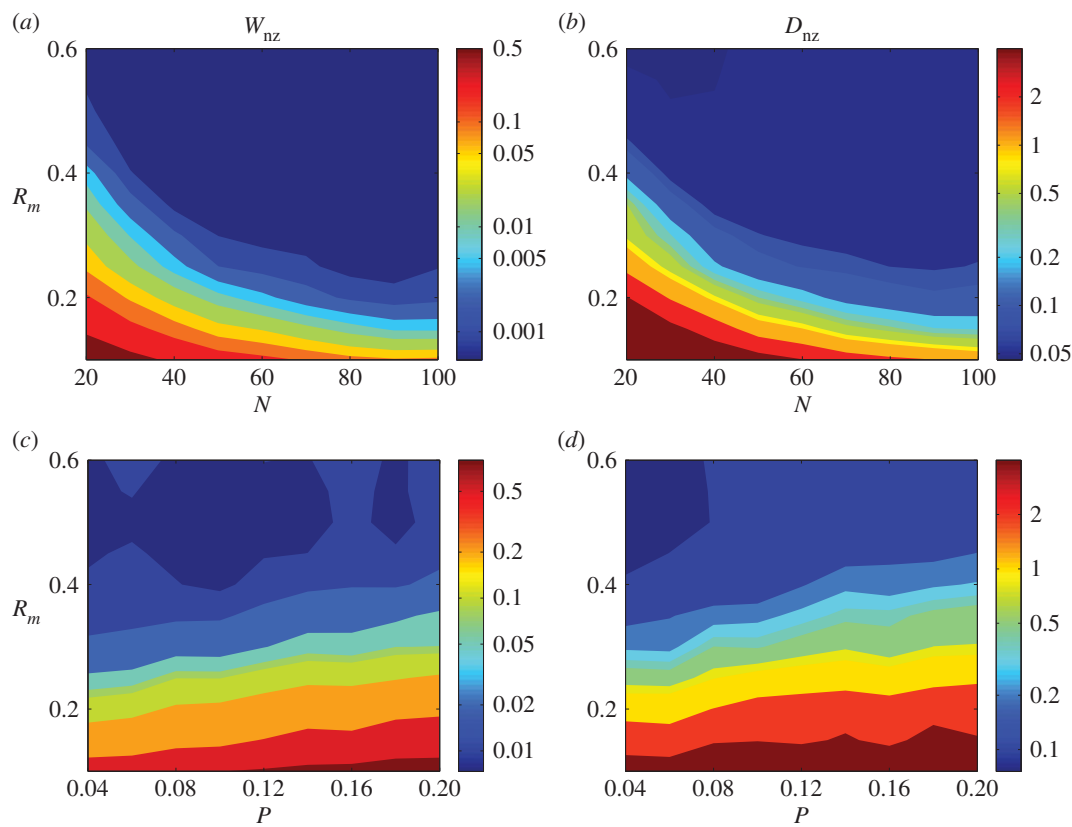
As the connection probability  $P$  is increased, for example, from 0.02 to 0.2, for a fixed network size (e.g.  $N = 30$ ), larger data amount is required for reliable reconstruction, as shown in figure 5c,d. This is consistent with previous results on reconstruction of complex networks without time delays [35,36], a feature of the compressive-sensing-based method.

### 2.3. Error analysis of triangulation algorithm for node positioning in geophysical space

To locate all nodes in a two-dimensional space requires knowledge of the positions of at least three nodes (minimally four nodes in a three-dimensional space). Due to noise, the required number of beacon nodes will generally be larger. Since node positioning is based on time delays estimated from compressive sensing, which contain errors, the number of required beacon nodes is larger than three even in two dimensions. To quantify the positioning accuracy, we use the normalized error  $M_r$ , defined as the medium distance error between the estimated and actual locations for all nodes (except the beacon nodes), normalized by the distributed length  $L$ . Figure 6 shows  $M_r$  versus the fraction  $R_B$  of the beacon nodes. The reconstruction parameters are chosen such that the errors in the time delay estimation is  $D_{nz} \approx 0.12$ . For small values of  $R_B$ , the positioning errors are large. Reasonable positioning errors are obtained when  $R_B$  exceeds, say, 0.2.



**Figure 4.** Effect of the amount of time delay on reconstruction performance. For networks with uniform time delays, errors of predicted weights (a) and delays (b) versus the length of the actual time delay in the network. Shown in (a) are the normalized errors associated with the non-zero terms in the weights, for several values of  $R_m$ . In (b), the errors are associated with the time delays of the existent links. Random networks of size  $N = 100$  and connection probability of  $P = 0.04$  are used. The results are from three different time-series segments, as marked with different symbols. Each data point is the result of averaging over 10 network realizations.

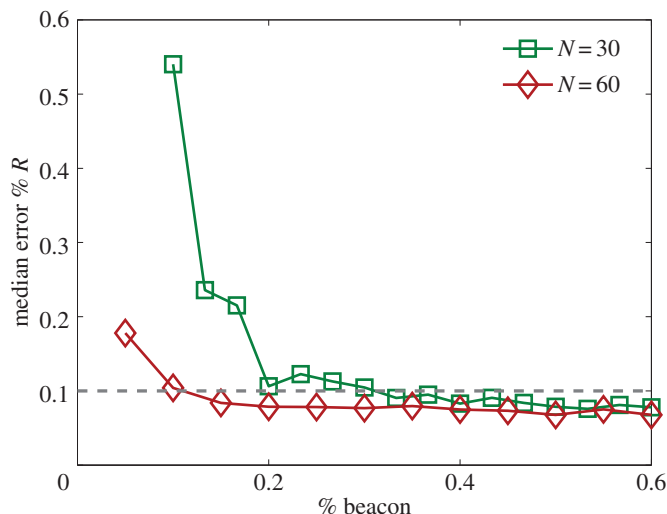


**Figure 5.** Effect of network size on reconstruction performance. (a,b) Errors associated with non-zero terms of weights  $W_{nz}$  and time delays  $D_{nz}$ , respectively, as the network size  $N$  is increased (left to right), for increasing data amount (bottom to top). The networks are random with fixed link probability  $p = 0.04$ . Cold colours represent small errors. (c,d) Errors in the weights (c) and delays (d) versus the connection probability  $p$  for fixed  $N = 30$ . All results are obtained by averaging over 20 independent network realizations.

### 2.4. Locating a hidden node in a geospatial network

To demonstrate that our compressive-sensing-based approach can be used to ascertain the existence of a hidden node and to estimate its physical location in a geospatial network, we use the model random network in figure 5. A node is regarded as ‘hidden’ when no time series or other type of direct information can be obtained from it. To detect a hidden node, it is necessary to identify its neighbouring





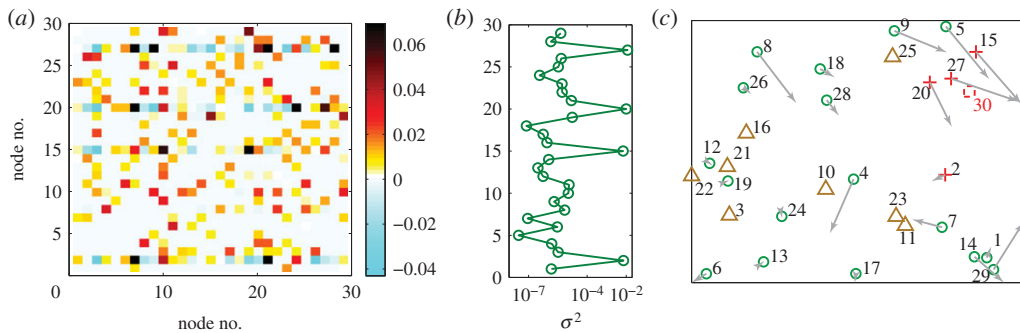
**Figure 6.** Positioning errors. Normalized positioning error  $M_r$ , defined as the medium absolute estimated distance error normalized by the distributed length  $L$ , as a function of the fraction  $R_b$  of the beacon nodes. The networks have the scale-free topology with the average outgoing degree  $k = 5$ . Two values of the network size are used:  $N = 30$  and  $N = 60$ . The beacon nodes are chosen as those having the largest degrees. The time delays are estimated using the data amount  $R_m = 0.5$ , for which the average error is  $D_{nz} \approx 0.12$ . The results are obtained by averaging over 10 independent network realizations.

nodes [38]. For an externally accessible node, if there is a hidden node in its neighbourhood, the corresponding entry in the reconstructed adjacency matrix will exhibit an abnormally dense pattern or contain meaningless values. In addition, the estimated coefficients for the dynamical and coupling functions of such an abnormal node typically exhibit much larger variations when different data segments are used, in comparison with those associated with normal nodes that do not have hidden nodes in their neighbourhoods. The mathematical formulation of our method to uncover a hidden node can be found in Methods. Initially, there are only 29 time series, one from each of the normal nodes, and it is not known *a priori* that there would be a hidden node in the network. We proceed to reconstruct the network to obtain the estimated weights and time delays, as shown in figure 7*a*. From the results, we find that the connection patterns of some nodes are relatively dense and the values of the weights and time delays are meaningless (e.g. negative values), giving the first clue that these nodes may be the neighbouring nodes of a hidden node. To confirm that this is indeed the case, we divide the available time series into a number of segments under the criterion that the data requirement for reconstruction is satisfied for each segment. As shown in figure 7*b*, we observe extraordinarily large variances in the estimated coefficients associated with the abnormal nodes. Combining results from figure 7*a,b*, we can claim with confidence that the four nodes are indeed in the immediate neighbourhood of a hidden node, ascertaining its existence in the network. Our method also works if there are more than one hidden node, given that they do not share common neighbouring nodes.

While the results from figure 7*a,b* confirm the existence of a hidden node in the network, its geophysical location is still unknown. Note that each of the neighbouring nodes of the hidden node is connected to a number of ‘normal’ nodes in the network. We can then use the standard triangularization procedure to determine the locations of all the ‘abnormal’ nodes. Since a geospatial random network has the property that nodes tend to connect with physically nearby nodes, we can deduce that the hidden node must be in the geographical vicinity of the abnormal nodes. In the example in figure 7, the hidden node (30, represented by red square) must then stay near its neighbouring nodes (nodes 2, 15, 20 and 27, represented by red crosses), as shown in figure 7*c*, where the normal nodes that are not in the neighbourhood of the hidden node are denoted by green circles.

### 3. Discussion

Given that data are available from a large number of components of a complex networked system which are distributed in a geophysical space, can the network structure be reconstructed, the locations of all nodes be determined and hidden nodes be detected and ascertained? We address these related issues



**Figure 7.** Detection of hidden nodes in geospatial networks. For a random network of  $N = 30$  nodes, illustration of detecting a hidden node (30). (a) Reconstructed time delays using time series from 29 externally accessible nodes. (b) Average variance in the reconstructed incoming coupling delays calculated from different segments of the available time series. (c) Estimated positions of all accessible nodes in comparison with the respective actual positions, and the location of the hidden node. The triangles denote the beacon nodes, whose positions are known *a priori*. The green circles denote ‘normal’ nodes without any hidden node in their immediate neighbourhoods, while the crosses are direct neighbours of the hidden node. The actual position of the hidden node 30 is marked as a dashed square.

by developing a compressive-sensing-based approach. In particular, assume that time series or signals from nodes in the network are collected at a single location. Our approach enables not only the network topology to be reconstructed, but also the various time delays of the signals from distinct nodes to be estimated. A standard triangularization procedure can then be used to determine the locations of the nodes in the geospatial network, in two-dimensional or three-dimensional space, based on the time delay estimates. We also demonstrate that the existence of a hidden node, from which no signal or time series is externally accessible, can be inferred and ascertained by identifying all nodes in its immediate neighbourhood. The location of the hidden node can then be estimated, as nodes in a geospatial network tend to be locally connected.

We stress that, for data-based reconstruction of complex geospatial networks, a significant challenge is that the available time series are time delayed, due to the finite speed of the physical signals. One unique contribution of this work, which goes beyond those of previous works on compressive-sensing-based reconstruction of complex networks [35–39,41], is a demonstration that inhomogeneous time delays in a complex network can be estimated reliably using compressive sensing. The information about the time delays allows us to determine the geophysical locations of the nodes. Our approach sheds new light into the general problem of network reconstruction and has potential applications in understanding and exploiting geographically embedded networks.

While we used continuous-time, oscillatory dynamics on networks as a prototypical model, our compressive-sensing-based framework for reconstruction of geospatial networks can be generalized to other types of network dynamical processes. For example, previous works demonstrated that both discrete time maps [35] and evolutionary game dynamics [37] can be formulated as an optimization problem that can be solved by compressive sensing. To include time delays for discrete time maps is straightforward. For evolutionary game dynamics, one can take into account time delays by using a delayed vector for each agent, the entries of which correspond to the time delays between this agent and all other agents in the network. For large networks, the required computation may be demanding. A more serious difficulty may arise when the delays among the agents are substantial, leading to the violation of the sparsity condition required for compressive sensing. To accurately estimate the time delays for complex networks hosting evolutionary game dynamics is thus an open problem at present.

There are still many challenges to apply our method to more general systems. For example, for the linear coupling scheme, we used the Taylor expansion to approximate the delayed coupling functions to the first order in the time delays so that its values can be estimated directly. However, if the coupling function is nonlinear, applying the Taylor expansion will result in complicated terms that involve high orders of the time delays, making extrapolating their values difficult (if not impossible). How to treat nonlinear coupling functions that contain time delays remains to be an open problem. Also, our method can locate nodes only in the Cartesian space, as the distances are inferred from the time delays information through a triangulation algorithm that is specifically designed to deal with the Cartesian space. Extending the methodology to curved space is an interesting issue but it is open at present. Another difficulty is that high-dimensional nodal dynamical equations may not be sparsely expressed

through a Taylor expansion. In this case, we can either choose alternative expansion techniques, such as Fourier series, or test alternative observation variables to obtain suitable response functions. For example, in the particular case of reconstructing stochastic evolutionary game networks [37,38], the dynamical processes are not described by nonlinear differential equations but are defined in terms of a set of stochastic game rules. At every time step, each agent chooses its strategy in a random manner. A workable set of observation variables can be the pay-offs of all the agents. It was demonstrated [37,38] that the network structure under stochastic game rules can still be constructed with high accuracy, based on compressive sensing.

## 4. Methods

### 4.1. Mathematical framework for reconstructing coupled oscillator networks with time delay

As proof of principle, we present our reconstruction framework using continuous time oscillator networks. There are  $N$  oscillators in the network, and the dynamical process on each node is described by a set of coupled ordinary differential equations. (A similar framework can be formulated for other types of dynamical processes, such as evolutionary games [37].) The  $m$ -dimensional state variable of node  $i$  is written as equation (2.2). To derive equation (2.3), we assume linear coupling functions and causality so that all  $\tau_{ij}$  ( $i, j = 1, \dots, N$ ) are positive (for simplicity). We regroup all terms directly associated with node  $i$  into  $F'_i[\mathbf{x}_i(t)]$ , where

$$F'_i[\mathbf{x}_i(t)] \equiv \mathbf{F}_i[\mathbf{x}_i(t)] - \mathbf{x}_i(t) \cdot \sum_{j=1, j \neq i}^N \mathbf{W}_{ij}, \quad (4.1)$$

and we expand  $F'_i[\mathbf{x}_i(t)]$  into the following series form:

$$F'_i[\mathbf{x}_i(t)] = \sum_{\gamma} \tilde{\alpha}^{(\gamma)} \cdot \tilde{\mathbf{g}}^{(\gamma)}[\mathbf{x}_i(t)], \quad (4.2)$$

where  $\tilde{\mathbf{g}}^{(\gamma)}[\mathbf{x}_i(t)]$  represents a suitably chosen set of orthogonal and complete base functions such that the coefficients  $\tilde{\alpha}^{(\gamma)}$  are sparse. To proceed, we approximate  $\mathbf{x}_j(t - \tau_{ij})$  as

$$\mathbf{x}_j(t - \tau_{ij}) \approx \mathbf{x}_j(t) - \tau_{ij} \dot{\mathbf{x}}_j(t), \quad (4.3)$$

so all the coupling terms with inhomogeneous time delays associated with node  $i$  can be written as

$$\left[ \sum_{j=1, j \neq i}^N \mathbf{W}_{ij} \mathbf{x}_j(t - \tau_{ij}) \right]_p \equiv \sum_{j=1, j \neq i}^N [\mathbf{b}_{ij} \mathbf{x}_j(t) + \mathbf{c}_{ij} \dot{\mathbf{x}}_j(t)], \quad (4.4)$$

where  $\mathbf{b}_{ij} = \mathbf{W}_{ij}$  and  $\mathbf{c}_{ij} = -\mathbf{W}_{ij} \tau_{ij}$ . Equation (2.2) can then be written in the compact form as equation (2.3), which is a set of linear equations, where  $\tilde{\alpha}^{(\gamma)}$ ,  $\mathbf{b}_{ij}$  and  $\mathbf{c}_{ij}$  are to be determined. If the unknown coefficient vectors can be reconstructed accurately, we will have complete information about the nodal dynamics as represented by  $F'[\mathbf{x}(t)]$ , the topology and interacting weights of the underlying network as represented by  $\mathbf{W}_{ij}$ , as well as the time delays associated with the non-zero links because of the relations  $\mathbf{W}_{ij} = \mathbf{b}_{ij}$  and  $\tau_{ij} = -\mathbf{c}_{ij}/\mathbf{b}_{ij}$ . However, if the coupling form is nonlinear, the relationship between the delay term  $\mathbf{c}_{ij}$  and the time delays  $\tau_{ij}$  would be hard to interpret, especially when the exact coupling form is not known.

As an illustrative example, we consider the case where each individual nodal dynamical system is three-dimensional with variables  $x$ ,  $y$  and  $z$ . For the first component  $x_i$  of node  $i$ , we have the following series expansion at time  $t$ :

$$\begin{aligned} \dot{x}_i = & (\tilde{a}_i)_{000} \cdot x_i^0 y_i^0 z_i^0 + \dots + (\tilde{a}_i)_{333} \cdot x_i^3 y_i^3 z_i^3 + (b_{i1})_1 x_1 + (b_{i1})_2 y_1 + (b_{i1})_3 z_1 + \dots + (b_{iN})_1 x_N \\ & + (b_{iN})_2 y_N + (b_{iN})_3 z_N + (c_{i1})_1 \dot{x}_1 + (c_{i1})_2 \dot{y}_1 + (c_{i1})_3 \dot{z}_1 + \dots + (c_{iN})_1 \dot{x}_N + (c_{iN})_2 \dot{y}_N + (c_{iN})_3 \dot{z}_N, \end{aligned} \quad (4.5)$$

where  $b_{ii}$  and  $c_{ii}$  are excluded. The formula contains three parts: the power series of isolated nodal dynamics with coefficients  $\tilde{a}_i$ , terms of all other coupled nodes' variables with coefficients  $b_{ij}$ , and terms of derivatives of the coupled nodes as represented by  $c_{ij}$ . Assuming that measurements  $x_i(t)$ ,  $y_i(t)$  and

$z_i(t)$  at a set of time instants  $t_1, t_2, \dots, t_w$  are available, we write

$$\mathbf{A}_i(t) = [x_i(t)^0 y_i(t)^0 z_i(t)^0, \dots, x_i(t)^3 y_i(t)^3 z_i(t)^3], \quad (4.6)$$

$$\mathbf{B}_i(t) = [x_1(t), \dots, x_N(t), y_1(t), \dots, y_N(t), z_1(t), \dots, z_N(t)] \quad (4.7)$$

and

$$\mathbf{C}_i(t) = [\dot{x}_1(t), \dots, \dot{x}_N(t), \dot{y}_1(t), \dots, \dot{y}_N(t), \dot{z}_1(t), \dots, \dot{z}_N(t)] \quad (4.8)$$

to obtain a compact expression

$$\mathbf{X} = \mathbf{G} \cdot \mathbf{a}_i + \xi, \quad (4.9)$$

where

$$\mathbf{X} = \begin{pmatrix} \dot{x}_i(t_1) \\ \dot{x}_i(t_2) \\ \vdots \\ \dot{x}_i(t_w) \end{pmatrix} = \begin{pmatrix} \mathbf{A}_i(t_1) & \mathbf{B}_i(t_1) & \mathbf{C}_i(t_1) \\ \mathbf{A}_i(t_2) & \mathbf{B}_i(t_2) & \mathbf{C}_i(t_2) \\ \vdots & \vdots & \vdots \\ \mathbf{A}_i(t_w) & \mathbf{B}_i(t_w) & \mathbf{C}_i(t_w) \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{a}}_i \\ \mathbf{b}_i \\ \mathbf{c}_i \end{pmatrix} + \xi, \quad (4.10)$$

and  $\xi$  represents the error introduced by the series approximation.

In general, the connection pattern of a complex network is far sparser than the all-to-all coupling configuration, so typically most elements of  $[\tilde{\mathbf{a}}_i, \mathbf{b}_i, \mathbf{c}_i]^T$  are zero. In addition, the error  $\xi$  is small and can be regarded as a noise term.

## 4.2. Compressive sensing algorithm in the presence of noise

The compressive sensing algorithm can be used to solve a sparse vector  $\mathbf{a}$  from the ill-conditioned linear equation under noise:  $\mathbf{X} = \mathbf{G} \cdot \mathbf{a} + \xi$ , where  $\xi$  is a random process. Reliable recovery of the sparse vector  $\mathbf{a}$  can be achieved [29,30,33] by solving the following  $l_1$  regularization problem:

$$\min \|\mathbf{a}\|_{l_1}, \quad \text{subject to } \|\mathbf{G} \cdot \mathbf{a} - \mathbf{X}\|_{l_2} \leq \varepsilon, \quad (4.11)$$

where the  $l_1$  norm for a vector  $\mathbf{x}$  is defined as  $\|\mathbf{x}\|_{l_1} = \sum_{i=1}^n |x_i|$  (its  $l_2$  norm is  $\|\mathbf{x}\|_{l_2} = \sqrt{\sum_{i=1}^n |x_i|^2}$ ) and  $\varepsilon$  is a threshold value determined by the noise amplitude. The reconstructed vector  $\tilde{\mathbf{a}}$  lies within the range:  $\|\tilde{\mathbf{a}} - \mathbf{a}\| \leq C \cdot \varepsilon$ , where  $C$  is a constant.

In order to apply the compressive sensing algorithm to solve equation (4.10), it is necessary to normalize each column of the matrix  $\mathbf{G}$  by the  $L_2$  norm of that column:  $(\mathbf{G}')_{ij} = (\mathbf{G})_{ij}/L_2(j)$  with  $L_2(j) = \sqrt{\sum_{i=1}^M [(\mathbf{G})_{ij}]^2}$ . We thus have  $\mathbf{X} = \mathbf{G}' \cdot \mathbf{u}'$  with  $\mathbf{u}' = \mathbf{u}L_2$ . After  $\mathbf{u}'$  is determined through some standard compressive sensing algorithm, the coefficients  $\mathbf{u}$  are given by  $\mathbf{u}'/L_2$ . Substituting  $\mathbf{a}_i$ ,  $\mathbf{b}_{ij}$  and  $\mathbf{c}_{ij}$  back into equation (2.3), we obtain the nodal dynamics, coupling weights and the delays associated with the first dynamical variable of node  $i$ . For the remaining two variables for this node and all variables for other nodes in the network, a similar procedure can be followed.

## 4.3. Triangle localization method

Given the positions of  $k$  reference nodes (or beacon nodes)  $(x_k, y_k)$ , and their distances  $d_{i,1}, d_{i,2}, \dots, d_{i,k}$  to the target node  $i$ , we can calculate the position of node  $i$  using the triangular localization method [44], for  $k$  larger than the space dimension. In general, we will need to solve the least-squares optimization problem  $\mathbf{H} \cdot \mathbf{x}_i = \mathbf{b}$ , where  $\mathbf{x}_i = [x_i, y_i]^T$  is the position of node  $i$ , and  $\mathbf{H} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k]^T$  is the position vector corresponding to the set of beacon nodes, where  $\mathbf{b} = 0.5 \times [D_1, D_2, \dots, D_k]^T$  and  $D_k = d_{ik}^2 - y_k^2 + x_k^2$ .

To locate the positions of all nodes in the network, we start with a small set of beacon nodes whose actual positions are known and the distances associated with all links in the network. Initially, we can locate the nodes that are connected to at least three nodes in the set of beacon nodes, insofar as the three reference nodes are not located on a straight line. When this is done, the newly located nodes can be added into the set of reference nodes and the neighbouring nodes can be located through the new set of beacon nodes. We iterate this process until the positions of all nodes are determined or no more qualified neighbouring nodes can be found. For a general network, such initial beacon sets may not be easily found. A special case is scale-free networks, for which the initial beacon set can be chosen as the nodes with the largest degrees. For a random network, we can also choose the nodes of the largest degree as the initial beacon node set and use a larger beacon set to locate most of the nodes in the network. For an arbitrary topology, we offer the following simple method to select the set of beacon nodes: we estimate the distances from one node to all other unconnected nodes using the weighted shortest distance and then proceed with the triangular localization algorithm. There are alternative localization algorithms based on given distances, for example, the multidimensional scaling method [52,53].

## 4.4. Asynchronous data collection

In real applications, the requirement to collect time series simultaneously will usually not be met. Consider the typical situation where the data are collected from a fixed external node, denoted by  $s$ , which has varying distances to the signal sources. The signals that arrive at the external node at time  $t$  were actually sent out by the sources at time  $t - \tau_{is}$ , where  $\tau_{is}$  is the varying transmission delay associated with the distances from node  $i$  to  $s$ . In general,  $\tau_{is}$  is unknown *a priori* because the location of node  $i$  needs to be determined.

In the reconstruction of the dynamical process and connections associated with node  $i$ , the time series substituted into equation (2.3) are in fact  $x_i(t - \tau_{is})$  and  $x_j(t - \tau_{js})$ , for  $j = 1, 2, \dots, N$  and  $j \neq i$ . The delay coupling terms can be approximated as

$$\mathbf{x}_j(t - \tau_{is} - \tau_{ij}) = \mathbf{x}_j(t - \tau_{js} - \tau_{is} - \tau_{ij} + \tau_{js}) \approx \mathbf{x}_j(t - \tau_{js}) - \tau'_{ij} \dot{\mathbf{x}}_j(t - \tau_{js}), \quad (4.12)$$

where  $\tau_{ij}$  is the actual delay and  $\tau'_{ij}$  is the estimated delay. From Taylor expansion, we have  $\tau'_{ij} = \tau_{is} - \tau_{js} + \tau_{ij}$ . Similarly, for node  $j$ , the estimated delay is  $\tau'_{ji} = \tau_{js} - \tau_{is} + \tau_{ji}$ . Because  $\tau_{ij} = \tau_{ji}$ , we can eliminate the effect of  $\tau_{is}$  and  $\tau_{js}$  by averaging the two estimated delays, as  $\tau_{ij} = \tau_{ji} = (\tau'_{ij} + \tau'_{ji})/2$ . After we obtain the time-delay matrix  $\{\tau'_{ij}\}$ , we can convert its elements into the actual elements so as to obtain the corresponding distances  $\{d_{ij}\}$ .

## 4.5. Locating a hidden node in a random geospatial network

We first reconstruct the network using our compressive sensing framework, treating the system as if there was no hidden node. As demonstrated in figure 7, the neighbouring nodes of the hidden node tend to exhibit abnormally dense connection patterns. We then use multiple time-series segments to calculate the variance in the reconstructed coefficient vectors for all nodes. In particular, the variance  $\sigma_i$  associated with node  $i$  is defined as

$$\sigma_i = \sqrt{\frac{1}{T} \sum_{t=1}^T \frac{1}{N} \sum_{k=1}^N (w_{ik} - \tilde{w}_{ik})^2}, \quad (4.13)$$

where  $T$  is the number of data segments used,  $N$  the network size and  $\tilde{w}_{ij}$  the average weights over  $T$  realizations. The variance associated with the time delays can be calculated in a similar way. The neighbouring nodes of the hidden node are those with abnormally dense connection patterns and significantly larger variances than others.

If there are more than one hidden node but they are not directly linked, i.e. they do not share any neighbouring nodes, we can use the cancellation method developed in our previous work [39] to ascertain the number of hidden nodes. The first step is to identify all neighbouring nodes. The second step is to check if the neighbouring nodes are connected to the same hidden node by calculating the cancellation factor for any pair of neighbouring nodes. If they are connected to the same hidden node, the cancellation factor is the ratio of their coupling strength to the common hidden node, which is typically non-zero; otherwise they are connected to different hidden nodes. The third step is to repeat this process for all possible pairs of the identified neighbouring nodes. This procedure allows us to classify the neighbouring nodes into different groups, each containing a hidden node.

**Authors' contributions.** Y.C.L. and W.X.W. devised the research project. R.Q.S. performed numerical simulations. R.Q.S., W.X.W., X.W. and Y.C.L. analysed the results. R.Q.S. and Y.C.L. wrote the paper. All authors gave final approval for publication.

**Competing interests.** The authors declare no competing financial interests.

**Funding.** This work was supported by ARO under grant no. W911NF-14-1-0504.

## References

- Gruen S, Diesmann M, Aertsen A. 2002 Unitary events in multiple single neuron spiking activity. I. Detection and significance. *Neural Comp.* **14**, 43–80. (doi:10.1162/089976602753284455)
- Gütig R, Aertsen A, Rotter S. 2002 Statistical significance of coincident spikes: count-based versus rate-based statistics. *Neural Comp.* **14**, 121–153. (doi:10.1162/089976602753284473)
- Gardner TS, di Bernardo D, Lorenz D, Collins JJ. 2003 Inferring genetic networks and identifying compound mode of action via expression profiling. *Science* **301**, 102–105. (doi:10.1126/science.1081900)
- Pipa G, Grün S. 2003 Non-parametric significance estimation of joint-spike events by shuffling and resampling. *Neurocomputing* **52**, 31–37. (doi:10.1016/S0925-2312(02)00823-8)
- Brovelli A, Ding M, Ledberg A, Chen Y, Nakamura R, Bressler SL. 2004 Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by Granger causality. *Proc. Natl Acad. Sci. USA* **101**, 9849–9854. (doi:10.1073/pnas.0308538101)
- Bongard J, Lipson H. 2007 Automated reverse engineering of nonlinear dynamical systems. *Proc.*



- Natl Acad. Sci. USA* **104**, 9943–9948. (doi:10.1073/pnas.0609476104)
7. Timme M. 2007 Revealing network connectivity from response dynamics. *Phys. Rev. Lett.* **98**, 224101. (doi:10.1103/PhysRevLett.98.224101)
  8. Tang WKS, Yu M, Kocarev L. 2007 Identification and monitoring of biological neural network. In *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, pp. 2646–2649. Piscataway, NJ: IEEE. (doi:10.1109/ISCAS.2007.377957)
  9. Napolitani D, Sauer TD. 2008 Reconstructing the topology of sparsely connected dynamical networks. *Phys. Rev. E* **77**, 026103. (doi:10.1103/PhysRevE.77.026103)
  10. Sontag E. 2008 Network reconstruction based on steady-state data. *Essays Biochem.* **45**, 161–176. (doi:10.1042/bse0450161)
  11. Wang WX, Chen QF, Huang L, Lai YC, Harrison M. 2009 Scaling of noisy fluctuations in complex networks and applications to network prediction. *Phys. Rev. E* **80**, 016116. (doi:10.1103/PhysRevE.80.016116)
  12. Ren J, Wang WX, Li B, Lai YC. 2010 Noise bridges dynamical correlation and topology in coupled oscillator networks. *Phys. Rev. Lett.* **104**, 058701. (doi:10.1103/PhysRevLett.104.058701)
  13. Yuan Y, Stan GB, Warnick S, Goncalves J. 2010 Robust dynamical network reconstruction. In *49th IEEE Conf. on Decision and Control (CDC)*, pp. 810–815. Piscataway, NJ: IEEE. (doi:10.1109/CDC.2010.5717657)
  14. Levnajić Z, Pikošky A. 2011 Network reconstruction from random phase resetting. *Phys. Rev. Lett.* **107**, 034101. (doi:10.1103/PhysRevLett.107.034101)
  15. Hempel S, Koseska A, Kurths J, Nikoloski Z. 2011 Inner composition alignment for inferring directed networks from short time series. *Phys. Rev. Lett.* **107**, 054101. (doi:10.1103/PhysRevLett.107.054101)
  16. Shandilya SG, Timme M. 2011 Inferring network topology from complex dynamics. *New J. Phys.* **13**, 013004. (doi:10.1088/1367-2630/13/1/013004)
  17. Yu D, Parlitz U. 2011 Inferring network connectivity by delayed feedback control. *PLoS ONE* **6**, e24333. (doi:10.1371/journal.pone.0024333)
  18. Pan W, Yuan Y, Stan GB. 2012 Reconstruction of arbitrary biochemical reaction networks: a compressive sensing approach. In *51st IEEE Annual Conf. on Decision and Control (CDC)*, pp. 2334–2339. Piscataway, NJ: IEEE. (doi:10.1109/CDC.2012.6426216)
  19. Wang WX, Ren J, Lai YC, Li B. 2012 Reverse engineering of complex dynamical networks in the presence of time-delayed interactions based on noisy time series. *Chaos* **22**, 033131. (doi:10.1063/1.4747708)
  20. Berry T, Hamilton F, Peixoto N, Sauer T. 2012 Detecting connectivity changes in neuronal networks. *J. Neurosci. Methods* **209**, 388–397. (doi:10.1016/j.jneumeth.2012.06.021)
  21. Stetter O, Battaglia D, Soriano J, Geisel T. 2012 Model-free reconstruction of excitatory neuronal connectivity from calcium imaging signals. *PLoS Comput. Biol.* **8**, e1002653. (doi:10.1371/journal.pcbi.1002653)
  22. Hamilton F, Berry T, Peixoto N, Sauer T. 2013 Real-time tracking of neuronal network structure using data assimilation. *Phys. Rev. E* **88**, 052715. (doi:10.1103/PhysRevE.88.052715)
  23. Zhou D, Xiao Y, Zhang Y, Xu Z, Cai D. 2013 Causal and structural connectivity of pulse-coupled nonlinear networks. *Phys. Rev. Lett.* **111**, 054102. (doi:10.1103/PhysRevLett.111.054102)
  24. Timme M, Casadiego J. 2014 Revealing networks from dynamics: an introduction. *J. Phys. A. Math. Theor.* **47**, 343001. (doi:10.1088/1751-8113/47/34/343001)
  25. Donges J, Zou Y, Marwan N, Kurths J. 2009 The backbone of the climate network. *Europhys. Lett.* **87**, 48007. (doi:10.1209/0295-5075/87/48007)
  26. Chan J, Holmes A, Rabadan R. 2010 Network analysis of global influenza spread. *PLoS Comput. Biol.* **6**, e1001005. (doi:10.1371/journal.pcbi.1001005)
  27. Makarov VA, Panetos F, Feo O. 2005 A method for determining neural connectivity and inferring the underlying network dynamics using extracellular spike recordings. *J. Neurosci. Methods* **144**, 265–279. (doi:10.1016/j.jneumeth.2004.11.013)
  28. Yao C, Bollt EM. 2007 Modeling and nonlinear parameter estimation with Kronecker product representation for coupled oscillators and spatiotemporal systems. *Phys. D* **227**, 78–99. (doi:10.1016/j.physd.2006.12.006)
  29. Donoho D. 2006 Compressed sensing. *IEEE Trans. Info. Theory* **52**, 1289–1306. (doi:10.1109/TIT.2006.871582)
  30. Candès E, Romberg J, Tao T. 2006 Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.* **59**, 1207–1223. (doi:10.1002/cpa.20124)
  31. Candès E. 2006 Compressive sampling. In *Proc. Int. Congress of Mathematicians*, vol. 3, pp. 1433–1452. Zürich, Switzerland: European Mathematical Society.
  32. Candès E, Wakin M. 2008 An introduction to compressive sampling. *IEEE Signal Process. Mag.* **25**, 21–30. (doi:10.1109/MSP.2007.914731)
  33. Candès E, Romberg J, Tao T. 2006 Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Info. Theory* **52**, 489–509. (doi:10.1109/TIT.2005.862083)
  34. Baraniuk RG. 2007 Compressed sensing. *IEEE Signal Process. Mag.* **24**, 118–121. (doi:10.1109/MSP.2007.4286571)
  35. Wang WX, Yang R, Lai YC, Kovanis V, Grebogi C. 2011 Predicting catastrophes in nonlinear dynamical systems by compressive sensing. *Phys. Rev. Lett.* **106**, 154101. (doi:10.1103/PhysRevLett.106.154101)
  36. Wang WX, Yang R, Lai YC, Kovanis V, Harrison MAF. 2011 Time-series-based prediction of complex oscillator networks via compressive sensing. *Europhys. Lett.* **94**, 48006. (doi:10.1209/0295-5075/94/48006)
  37. Wang WX, Lai YC, Grebogi C, Ye JP. 2011 Network reconstruction based on evolutionary-game data via compressive sensing. *Phys. Rev. X* **1**, 021021. (doi:10.1103/PhysRevX.1.021021)
  38. Su RQ, Wang WX, Lai YC. 2012 Detecting hidden nodes in complex networks from time series. *Phys. Rev. E* **85**, 065201. (doi:10.1103/PhysRevE.85.065201)
  39. Su RQ, Lai YC, Wang X, Do YH. 2014 Uncovering hidden nodes in complex networks in the presence of noise. *Sci. Rep.* **4**, 3944. (doi:10.1038/srep03944)
  40. Su RQ, Ni X, Wang WX, Lai YC. 2012 Forecasting synchronizability of complex networks from data. *Phys. Rev. E* **85**, 056220. (doi:10.1103/PhysRevE.85.056220)
  41. Shen ZS, Wang WX, Fan Y, Di ZR, Lai YC. 2014 Reconstructing propagation networks with natural diversity and identifying hidden source. *Nat. Commun.* **5**, 4323. (doi:10.1038/srep11404)
  42. Wu X. 2008 Synchronization-based topology identification of weighted general complex dynamical networks with time-varying coupling delay. *Phys. A* **387**, 997–1008. (doi:10.1016/j.physa.2007.10.030)
  43. Husmeier D. 2003 Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics* **19**, 2271–2282. (doi:10.1093/bioinformatics/btg313)
  44. Sayed A, Tarighat A, Khajehnouri N. 2005 Network-based wireless location: challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Process. Mag.* **22**, 24–40. (doi:10.1109/MSP.2005.1458275)
  45. Moon IC, Carley K. 2007 Modeling and simulating terrorist networks in social and geospatial dimensions. *IEEE Intell. Syst.* **22**, 40–49. (doi:10.1109/MIS.2007.4338493)
  46. Li C, Chen G. 2004 Synchronization in general complex dynamical networks with coupling delays. *Phys. A* **343**, 263–278. (doi:10.1016/j.physa.2004.05.058)
  47. Dhamala M, Jirsa V, Ding M. 2004 Enhancement of neural synchrony by time delay. *Phys. Rev. Lett.* **92**, 074104. (doi:10.1103/PhysRevLett.92.074104)
  48. Gross T, Rudolf L, Levin S, Dieckmann U. 2009 Generalized models reveal stabilizing factors in food webs. *Science* **325**, 747–750. (doi:10.1126/science.1173536)
  49. Barabási AL, Albert R. 1999 Emergence of scaling in random networks. *Science* **286**, 509–512. (doi:10.1126/science.286.5439.509)
  50. Erdős P, Rényi A. 1959 On random graphs I. *Publ. Math. Debrecen* **6**, 290–291.
  51. Bellen A, Zennaro M. 2013 *Numerical methods for delay differential equations*. Oxford, UK: Oxford University Press.
  52. Shang Y, Ruml W. 2004 Improved MDS-based localization. In *23rd Annual Joint Conf. of the IEEE Computer and Communications Societies*, vol. 4, pp. 2640–2651. Piscataway, NJ: IEEE. (doi:10.1109/INFCOM.2004.1354683)
  53. Shang Y, Ruml W, Zhang Y, Fromherz MP. 2003 Localization from mere connectivity. In *Proc. 4th ACM Int. Symp. on Mobile ad hoc Networking & Computing*, pp. 201–212. New York, NY: ACM. (doi:10.1145/778415.778439)