Distributed Detection in Sensor Networks: Connectivity Graph and Small World Networks

Saeed A. Aldosari and José M. F. Moura
Electrical and Computer Engineering Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213-3890, USA
Email: {saldosar, moura}@ece.cmu.edu

Abstract— We study distributed detection in a sensor network where the sensors cooperate by exchanging information to reach a common understanding about the environment. We address two main issues: (1) distributed fusion: how to achieve a global decision without transmitting the information (measurements or local decisions) from all the sensors to a common central location like in parallel architectures; and (2) connectivity graph: what should be the connectivity pattern among the sensors, in other words, with which sensors should each sensor communicate. This is a nontrivial question since it corresponds to designing the structure of a graph to achieve a given goal. For the first issue, we propose an iterative algorithm that fuses the data globally without the need for collecting them at one central location. For the second issue, we present a design methodology based on "small world" network engines that leads to connectivity patterns that provide fast convergence to the distributed detection algorithm. Results show that introducing 10% to 30% randomness in the connectivity graph leads to significant improvements over both regular patterns and totally random networks.

I. INTRODUCTION

An important issue in sensor networks is the design of the underlying connectivity network that supports the information exchange among sensors. We refer to this problem as the connectivity graph or pattern of the sensor network. This paper discusses a methodology to design the connectivity graph of a sensor network that supports efficient and good performing distributed decision making algorithms in the network. This is a difficult problem since it corresponds to the design of the structure of a graph rather than the easier problem of designing the weights of a graph with *pre-specified* structure. Our solution finds the "best" connectivity graph within a specified class of graphs that exhibit "small world" properties.

Parallel fusion networks, which have been studied extensively in the literature ([1], Ch. 3), assume global connectivity of all the sensors in the network to a central location. Parallel networks are unduly dependent on the reliable operation of a single fusion center. They also require direct linkage of all sensors to the same global location. In many applications, these are not justifiable assumptions and architectures that support distributed decision making are needed. In this paper, we consider two related important issues: an iterative distributed detection algorithm that, unlike parallel fusion, can handle arbitrary connectivity patterns among sensors; and the design of the structure of the graph representing the sensor network

connectivity pattern to guarantee fast convergence of the iterative distributed detection algorithm. To design the graph structure, we present a design methodology based on "small world" network engines that leads to connectivity patterns that provide fast convergence to the distributed detection algorithm. The connectivity graph is derived from a regular pattern by random rewiring of a few connections—we show that random rewiring of only a small fraction of the original connections, 10% to 30% leads to significant faster convergence over regular patterns or totally random networks.

We present the problem in the very simple context of binary hypothesis testing, in which the state of the environment takes one of two possible alternatives, H_0 (target absent) or H_1 (target present). The true state H is monitored by N sensors, which collect N real measurements $\mathbf{y} = (y_1, \ldots y_N)$. Given the true state H_i, the sensor measurements are assumed to be independent and identically distributed (i.i.d.) with known conditional density $f_i(y) = f(y|H_i)$, $i = 0, 1$. Sensors are not assumed to be fully connected but, rather, they communicate over a network described by an undirected graph $G = (V, E)$. The graph is specified by a set of N vertices $V = \{s_1, \ldots, s_N\}$, which correspond to sensor nodes, and a set of M edges $\mathcal{E} = \{l_m = (i_m, j_m), \ m = 1, \ldots, M\}$, where $(i, j) \in \mathcal{E}$ whenever sensor s_i can communicate with s_j. We assume the graph G to be *connected* in the sense that there is a *path* from every vertex s_i to every other vertex s_j, $i, j = 1, \ldots, N$.

The goal of distributed detection is for all sensor nodes to reach a global common decision $\widetilde{H}$ about the true state H based on the measurements collected by all sensors through exchange of information over the network G. We adopt a Bayesian framework for which the detection performance is measured by the average probability of decision error $P_e = \Pr(\widetilde{H} \neq H) = \pi_0 P_{e0} + \pi_1 P_{e1}$, where $P_{e0} = \Pr(\widetilde{H} = H_1|H_0)$, $P_{e1} = \Pr(\widetilde{H} = H_0|H_1)$, and π_0, π_1 are the prior probabilities of H_0 and H_1, respectively.

We focus on two main problems related to this distributed fusion task. First, how to achieve global fusion of the information collected by all sensors without having all data delivered to one central location and without requiring a fully connected network as is usually done in parallel architectures. Second, how does the connectivity pattern of the network affect the

overall detection performance. Specifically, for example, if the total number of communication links is constrained, should these links be arranged regularly to connect neighboring sensors or should they distribute randomly without any particular structure.

To address the first problem, we consider an iterative algorithm, which enables all sensor nodes to reach a consensus about the true hypothesis H through iterative exchange of information through the edges E of the network G. This distributed iterative algorithm does not require any routing mechanism since we do not need to collect the information at any one central location. Our design goal for this iterative algorithm is to maximize convergence speed because it is directly related to the communication cost. It turns out that convergence speed is intimately dependent on the connectivity pattern of the network G, which motivates our investigation of the second problem above. We adopt the convergence speed as performance measure to compare different connectivity structures. For simplicity, in this paper, we assume equal cost for all communication links, i.e., $\text{Cost}(l_i) = \text{Cost}(l_j)$, $i, j = 1, \ldots, M$, regardless of the *physical* length of these links. While such assumption may not hold for every wireless networks, it does accommodate many wireless and wired sensor network scenarios. In any case, results based on this cost assumption can serve as a first-cut study to have insight over what can be characterized as "good" connectivity patterns.

II. ITERATIVE DISTRIBUTED DETECTION

In this section, we assume that the structure of the connectivity graph of the sensor network has been fixed and is known. There still is the problem of determining the weights of this graph to speed the convergence of the iterative distributed detection algorithm. This is discussed in subsection II-B. Section III addresses the more difficult problem of *designing the structure* of the connectivity graph.

A. Parallel Fusion

To better understand the proposed iterative algorithm we first explain how the data are fused in parallel architectures. In the parallel fusion model, which has been extensively studied in the literature, all sensors deliver their measurements to one location known as the fusion center. This could be done through single or multi-hop links between every sensor and the fusion center. Once the data are collected at one location the task of fusing them to reach the decision with minimum Bayesian probability of error is a likelihood ratio test (LRT) of the form

$$\frac{f(\mathbf{y}|H_1)}{f(\mathbf{y}|H_0)} \overset{\widetilde{H}=1}{\underset{\widetilde{H}=0}{\gtrless}} \frac{\pi_0}{\pi_1}.$$

Due to the conditional independence of the samples $y_1, \ldots y_N$, the optimum fusion rule can be written as a function of the sum of local log likelihood ratios (LLRs) as follows

$$R = \sum_{n=1}^{N} r_n \overset{1}{\underset{0}{\gtrless}} \log \frac{\pi_0}{\pi_1} \triangleq \upsilon,$$

where R is the global LLR, $r_n = \log \frac{\Pr(y_n|H_1)}{\Pr(y_n|H_0)}$ is the local log likelihood ratio of the measurement of sensor n, and υ is the fusion threshold.

From this discussion, it is clear that the optimal fusion rule requires computing the sum of all the local statistics. This can be carried out by delivering all the data to a central location through direct or multihop communication links. Next, we propose a technique that accomplishes optimal fusion without routing all data to a single location.

B. Iterative Detection Procedure

We propose an iterative algorithm that does not require collecting all the information in a single central location. The scheme is based on a distributed linear averaging algorithm, which has been recently proposed in the context of estimation [2],[3]. It does not require any routing or flooding mechanism. Rather, the statistics stored at each sensor are updated in an iterative manner to reach a consensus about the global statistic. This updating is accomplished by exchanging the current statistics only with neighboring sensors.

Initially, sensors take measurements $y_1, \ldots y_N$, after which each sensor s_n, $n = 1, \ldots, N$ computes the log likelihood ratio $r_n = \log \frac{\Pr(y_n|H_1)}{\Pr(y_n|H_0)}$ of its measurement y_n. These LLR values serve as the initial states of the iterative algorithm. We denote the value stored at sensor s_n at iteration t by $x_n(t)$, and we refer to it as the *state* of s_n at time t. At $t = 0$, the initial states are given by $x_n(t = 0) = r_n$, $n = 1, \ldots, N$. The iterative procedure is then carried out according to the following linear operation [2]

$$x_n(t) = w_{nn}x_n(t-1) + \sum_{i \in \Omega_n}^{N} w_{ni}x_i(t-1), \qquad (1)$$

where Ω_n is the set of neighbors of sensor s_n and w_{ij} is a weight associated with edge (i, j) if this edge exists. The weight value is zero when there is no link associated with it, i.e., $w_{ij} = 0$ if $(i, j) \notin \mathcal{E}$. This can be expressed in matrix format as follows

$$\mathbf{x}(t) = \mathbf{W}\mathbf{x}(t-1), \qquad (2)$$

where $\mathbf{x}(t)$ is an $N \times 1$ vector of all current states and $\mathbf{W} = \{w_{ij}\}$ is a matrix of all weights. The iterative procedure is carried out until $x_n(t)$ of all sensors converge to the global average log likelihood ratio R/N. The final decision is then obtained locally at each sensor according to its current state

$$x_n(t) \overset{\widetilde{H}(n)=1}{\underset{\widetilde{H}(n)=0}{\gtrless}} \upsilon/N,$$

where $\widetilde{H}(n)$ denotes the decision of sensor s_n. Note that the updating rule (2) can be written in terms of the initial states as follows

$$\mathbf{x}(t) = \mathbf{W}^t \mathbf{x}(0), \qquad (3)$$

where $\mathbf{x}(0)$ is an $N \times 1$ vector of all the initial states $x_n(0)$, $n = 1, \ldots, N$.

C. Weight Design

The goal of the iterative algorithm proposed above is for every sensor state $x_n(t)$ to converge to the global average LLR $R/N = \frac{1}{N}\sum_{n=1}^{N} r_n$. This is done by designing the weight matrix W to ensure and accelerate the convergence of the iterative procedure. From (3), this is equivalent to requiring that W^t converge to the scaled all-ones matrix, i.e., $\lim_{t\to\infty} W^t = \frac{1}{N}\mathbf{1}_{N\times N}$. For a fixed network topology G, the issue of optimal design of the weights W has been treated in [2] and [3]. Several techniques have been proposed and their performance have been compared. One of the simplest techniques from the point of view of practical implementation is to assign an equal weight α to all network links. The weight associated with a link (i,j) is given by

$$w_{ij} = \begin{cases} \alpha & \text{if } (i,j) \in \mathcal{E} \\ 1 - \alpha d_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}, \qquad (4)$$

where $d_i = |\Omega_n|$ is the degree of sensor node i, i.e., the number of sensors connected to sensor s_i. This can be written in matrix form as $W = I_N - \alpha L$, where I_N is the $N \times N$ identity matrix and L is the Laplacian matrix associated with the graph G, the elements of which are given by

$$l_{ij} = \begin{cases} -1 & \text{if } (i,j) \in \mathcal{E} \\ d_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}.$$

The optimum choice of α that minimizes the spectral radius of $(W - \frac{1}{N}\mathbf{1}_{N\times N})$ and, hence, accelerates the convergence is given by [2]

$$\alpha^* = \frac{2}{\lambda_1(L) + \lambda_{N-1}(L)}, \qquad (5)$$

where $\lambda_1(L) > \cdots > \lambda_{N-1}(L) > 0$ are the ordered eigenvalues of L. For proofs and other weight design techniques the reader is referred to [2] and [3].

D. Performance Analysis

We analyze the performance of the proposed iterative consensus detection algorithm. To facilitate the analysis, we consider a Gaussian shift-in-mean model for which the observation at sensor s_n under H_i is characterized by

$$\text{under } H_i : y_n = \mu_i + z, \; z \sim \mathcal{N}(0, \sigma^2)$$

where μ_i is the signal mean under H_i, $i = 0, 1$ and z is a zero-mean Gaussian noise with variance σ^2. Without loss of generality, we let $\mu_1 = -\mu_0 = \mu$. Under this model it can be shown that the local LLRs r_n are also Gaussian distributed, i.e.,

$$\text{under } H_i : r_n \sim \mathcal{N}(\frac{2\mu\mu_i}{\sigma^2}, \frac{4\mu^2}{\sigma^2})$$

Before analyzing the iterative algorithm, let us first consider the performance of the parallel fusion architecture. In a centralized scheme, the decision is simply based on R, which is a sum of N i.i.d. variables r_n. Due to the conditional independence assumption, this sum is Gaussian with mean and variance given by

$$\text{under } H_i : R \sim \mathcal{N}(\frac{2\mu\mu_i}{\sigma^2}N, \frac{4\mu^2}{\sigma^2}N).$$

The probability of error is given by the tail distribution of R, i.e.,

$$\begin{aligned} P_{e1}^* &= Q\left(-\frac{\upsilon - 2\mu^2 N/\sigma^2}{2\mu\sqrt{N}/\sigma}\right) \\ P_{e0}^* &= Q\left(\frac{\upsilon + 2\mu^2 N/\sigma^2}{2\mu\sqrt{N}/\sigma}\right), \end{aligned} \qquad (6)$$

where $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$ is the normal right-tail distribution. The last equation provides the minimum error probabilities achieved by a parallel fusion network, which can serve as a lower bound for other fusion techniques.

In the proposed iterative algorithm (2), decisions are made at all sensor nodes based on a weighted sum of the measured LLRs r_n, $n = 1, \ldots, N$. In the limit, when the number of iterations t go to ∞, the weight matrix W^t converges to $\frac{1}{N}\mathbf{1}_{N\times N}$ and therefore, the detection performance approaches that of the optimum centralized scheme in (6). It can be shown that at iteration t, the probability of error of the sensor s_n is given by

$$\begin{aligned} P_{e1}^n &= Q\left(-\frac{\upsilon - 2\mu^2 N/\sigma^2}{2\mu N\sqrt{[W^{2t}]_{ii}}/\sigma}\right), \\ P_{e0}^n &= Q\left(\frac{\upsilon + 2\mu^2 N/\sigma^2}{2\mu N\sqrt{[W^{2t}]_{ii}}/\sigma}\right), \end{aligned} \qquad (7)$$

where $[W^{2t}]_{ii}$ denotes the ith diagonal element of W^{2t}. As $t \to \infty$, $[W^{2t}]_{ii}$ converges to $1/N$ and (7) approaches (6).

E. Numerical Examples

We present numerical examples to illustrate the operation of the iterative detection algorithm and to motivate the discussion in the following section. We consider two simple 5-sensor networks with circular and linear topologies as shown in Fig. 1. These simple cases are intended to illustrate the idea and simplify the discussion but the algorithm itself is in no way limited to such simple scenarios. In the following section, we deal with more complex examples involving large number of sensors and random topologies.

We let the signal strength be $\mu = 1$ while the noise variance $\sigma^2 = 0.501$, which results in a signal-to-noise ratio of SNR= 3dB, where the SNR in dB is given by $10\log_{10}(\mu^2/\sigma^2)$. We also assume equal priors for the state H, i.e., $\pi_0 = \pi_1 = 1/2$, which leads to a zero optimum fusion threshold $\upsilon = 0$.

In the proposed iterative algorithm the data are not delivered to one location. Instead, sensor nodes exchange and update their information in an iterative process till they reach a consensus about the monitored hypothesis H. Initially, all sensors $s_1, \ldots, s_5$ take measurements $y_1, \ldots, y_5$ and compute their LLRs $r_1, \ldots, r_5$, which serve as the initial states $x_1(0), \ldots, x_5(0)$ for the iterative algorithm. Each sensor, then, exchanges its stored state $x_n(0)$ with neighboring sensors
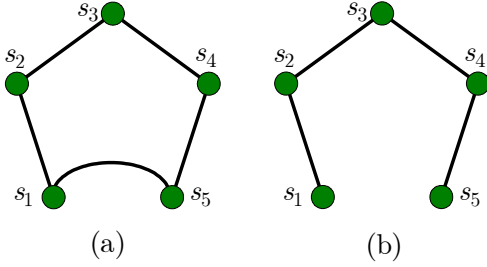
Fig. 1. A five node (a) circular graph, (b) linear graph.

Fig. 2. Convergence of the detection error of the iterative algorithm.

then updates its current state $x_n(1)$ according to (1). We use the optimal constant weights method described by (4). From (5), the optimum constant weights are found to be $\alpha^*_{\text{circular}} = 0.4$ and $\alpha^*_{\text{linear}} = 0.5$ for the circular and linear topologies, respectively. The iterative algorithm continues until the states $x_n(t)$, $n = 1, \ldots, 5$ simultaneously converge to the global average LLR $R/N = \frac{1}{5} \sum_{n=1}^{5} r_n$ or until a predefined maximum number of iterations T is reached. The sensor nodes then make their own decisions locally based on the last states $x_n(T)$. The probability of error for any sensor node s_n at any iteration t can be computed from (7). Fig. 2 shows the convergence of the error probability as the number of iterations is increased. Note that in this special example, all nodes perform equally but this may not be the case in general. It can be seen that the algorithm converges in both topologies to the optimum performance 7.93×10^{-4} (computed from (6)). However, there is a considerable difference in the convergence speed of the two topologies. For the linear topology, the algorithm converges in about 12 iterations which means a communication load of $(12)(5) = 60$ transmissions. On the other hand, by adding the shortcut between s_1 and s_5, the circular topology converges in 3 iterations resulting in a load of $(3)(5) = 15$ transmissions. It is interesting to see how a single shortcut can speed up the convergence by 4 times.

Aside from illustrating the basic idea, results demonstrate the impact of the connection pattern of the network on the performance of the iterative algorithm. Next, we elaborate more on this issue by studying the performance of the iterative detection algorithm as applied to a class of random networks.

III. CONNECTIVITY GRAPH DESIGN

The example in the previous section showed how the simple modification of the underlying connectivity pattern of the sensor network, obtained by adding a single shortcut, can dramatically improve the convergence speed of the iterative algorithm. In this section, we address the difficult problem of *designing* the *structure* of the connectivity graph of the sensor network. Rather than attacking this problem in general, we consider the optimization of the network within a given class of random networks that exhibit small-world behavior.
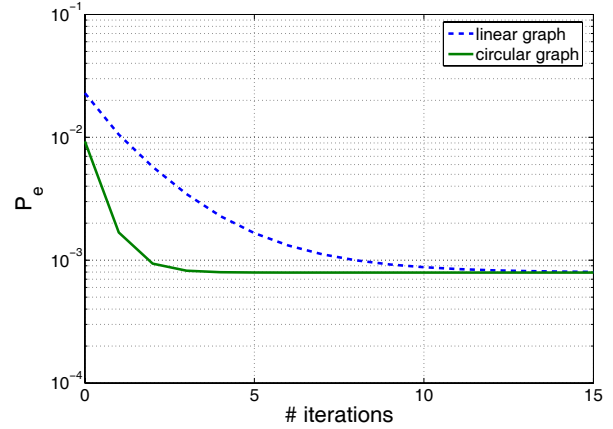
A. Small-World Networks

In many applications, it is commonly desirable to have networks that exhibit small diameter and strong local clustering. Regular networks (e.g., with nearest neighbor connectivity) are highly structured networks characterized by their strong clustering behavior but they suffer from their large diameter, which grows with the network size. On the other extreme, random networks are characterized by their small diameter but they have poor clustering. Small-world networks have small diameters close to that of random networks yet they are highly structured. Examples of such networks include the Internet, nervous system, US power grid, and social networks. The network diameter is measured by the *characteristic path length*, which is defined as the average path length between any pair of nodes. The clustering is quantified by a *clustering coefficient* defined as the fraction of neighbors of a node s_n that are also neighbors of each other.

In [4], Watts and Strogatz presented a formal model that enables the generation of graphs ranging from a completely random to a highly structured network. Also, it was shown through numerical simulation that in order for the network to exhibit a small-world phenomenon, it does not have to be highly random. In this section, we restrict our attention to this model to study the effect of adding random shortcuts on the behavior of the iterative detection algorithm.

In the Watts and Strogatz model [4], a graph is generated by first constructing a highly structured circular network as in Fig. 1 (a). However, all nodes are connected to their k nearest neighbors. Then a random rewiring procedure is conducted on all graph links. With probability ρ, a link is rewired to a different destination chosen uniformly at random. Notice that the ρ parameter controls the "randomness" of the graph in the sense that $\rho = 0$ corresponds to the original highly structured network while $\rho = 1$ results in a completely random network. Self and parallel links are prevented in the rewiring procedure while the number of links is always kept constant regardless of ρ.

B. Effect of Random Rewiring on Iterative Detection

Due to space limitation we only show results of one example. We study the effect of the connectivity pattern on the performance of the iterative algorithm. Due to their small diameter, one may expect that completely random networks outperform other structured networks. We investigate this issue through studying the convergence speed of the iterative detection algorithm on graphs generated using the Watts and Strogatz model.

We consider a class of networks with $N = 200$ sensors generated using the Watts and Strogatz model. Prior to rewiring, an initial circular graph is constructed where each node is connected with its $k = 3$ nearest neighbors. The probability ρ of rewiring is varied from 0 to 1. For each generated network, we apply the iterative detection algorithm and measure the average number of iterations $T(\rho)$ it takes to reach within 10% of the optimum error probability when the SNR is -10 dB. In addition, we evaluate the characteristic path length J and the average clustering coefficient C. Results are averaged over 1000 graphs generated using the Watts and Strogatz model. Fig. 3 shows the characteristic path length and the clustering coefficients normalized with that of regular networks ($\rho = 0$). The figure confirms the original findings Watts and Strogatz model [4] in that there is a range of ρ where the generated graphs exhibit a small-world phenomenon, i.e., high clustering coefficient and low path length. It also illustrates the fundamental difference between completely regular ($\rho = 0$) networks with high clustering coefficient and path length versus completely random networks ($\rho = 1$) having low path length but also poor clustering coefficient.

Fig. 4 shows the convergence speed of the iterative detection algorithm normalized with that of regular networks ($\rho = 0$). The figure exhibit a peak at $\rho = 0.3$, which clearly demonstrates the importance of having a "certain amount" of structure in the connectivity pattern. In other words, having a low characteristic length alone is not enough but it is also desirable to have a reasonable clustering coefficient. Fig. 4 also exhibit a *phase-change* phenomenon similar to that observed in [5]. In [5] the performance is quantified using the *algebraic connectivity* measure, which depends on the Laplacian matrix L of the graph. In contrast, we consider a measure that is directly related to detection applications. The phase-change in both cases occur around $\rho = 0.1$. It is interesting to see how the random rewiring procedure with small rewiring probability ρ can result in large speed-ups. When $\rho = 0.3$ the convergence speed is 35 times higher than that of regular networks and 2 times higher than a completely random network.

IV. CONCLUSIONS

We addressed two issues related to distributed detection in sensor networks. First, we proposed an iterative algorithm that fuses the information globally in a distributed manner without relying on any sort of centralized processing. This is crucial from the point of view of reliability and resource balancing. A single attack on the "fusion center" of a parallel architecture can bring the whole network down. In contrast,
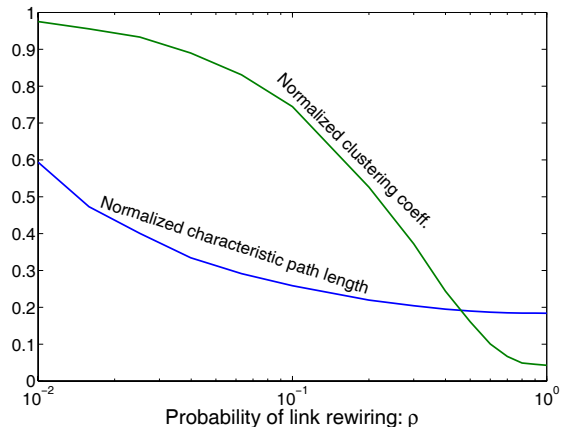

Fig. 3. Normalized characteristic path length ($J(\rho)/J(0)$) and normalized clustering coefficient ($C(\rho)/C(0)$)
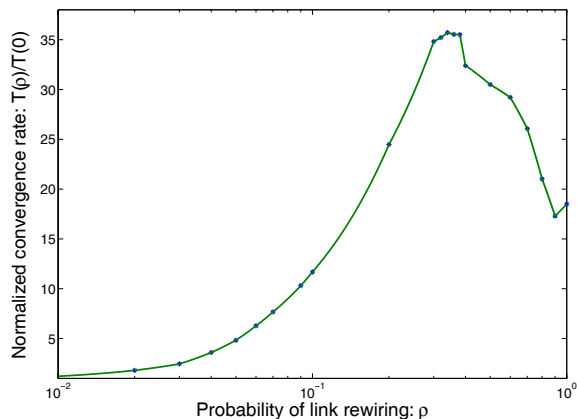

Fig. 4. Normalized convergence speed of the iterative detection algorithm for a range of small-world networks.

similar attacks (or faults) on the proposed fusion scheme will only cause a graceful degradation in performance. The second issue that we addressed is the design of the connectivity pattern to ensure fastest convergence of the iterative algorithm. Simulation results show that the best network structure within a class of small-world networks is neither completely random nor totally regular, which is surprising given the small characteristic length exhibited by completely random networks.

REFERENCES

[1] P. K. Varshney, *Distributed Detection and Data Fusion*. New York: Springer-Verlag, 1996.
[2] L. Xiao and S. Boyd, "Fast linear iteration for distributed averaging," *Syst. Contr. Lett.*, vol. 53, pp. 65–78, Sept. 2004.
[3] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automat. Contr.*, vol. 49, pp. 1520–1533, Sept. 2004.
[4] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, pp. 440–442, June 1998.
[5] R. Olfati-Saber, "Ultrafast consensus in small-world networks," in *Proc. of the American Control Conference*, Portland, OR, June 2005.