Information Processing Letters 115 (2015) 408–413

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

A greedy topology design to accelerate consensus in broadcast wireless sensor networks ☆

Massimo Vecchio [a,*], Roberto López-Valcarce [b]

[a] CREATE-NET, Via alla Cascata 56/D, 38123 Trento, Italy
[b] Departamento de Teoría de la Señal y las Comunicaciones, Universidad de Vigo, 36310 Vigo, Spain

ARTICLE INFO

Article history:
Received 19 February 2014
Received in revised form 18 November 2014
Accepted 18 November 2014
Available online 27 November 2014
Communicated by M. Chrobak

Keywords:
Average consensus
Range assignment
Topology control
Wireless multicast advantage
Graph algorithms

ABSTRACT

We present techniques to improve convergence speed of distributed average consensus algorithms in wireless sensor networks by means of topology design. A broadcast network is assumed, so that only the transmit power of each node can be independently controlled, rather than each individual link. Starting with a maximally connected configuration in which all nodes transmit at full power, the proposed methods successively reduce the transmit power of a chosen node in order to remove one and only one link; nodes are greedily selected either in order to yield fastest convergence at each step, or if they have the largest degree in the network. These greedy schemes provide a good complexity–performance tradeoff with respect to full-blown global search methods. As a side benefit, improving the convergence speed also results in savings in energy consumption with respect to the maximally connected setting.

1. Introduction

Average consensus, in the general framework of *networks of agents*, means reaching an agreement on the average state of all agents. Recently, much effort has been directed to the study of the average consensus problem in Wireless Sensor Networks (WSNs) (see [1] and the references therein), since distributed consensus algorithms only require iterative local information exchanges among neighboring nodes and the computation of weighted sums at each node. Potential applications include detection, estimation, reputation management, load balancing, control of autonomous agents, etc. [2].

One important issue regarding distributed average consensus algorithms in WSNs is convergence speed: reducing the convergence time results in fewer transmissions and therefore in energy savings. Approaches from the literature to speed up convergence can be classified into two groups. If the topology of the network is fixed, one can design the weights intervening in the consensus scheme in order to minimize convergence time [3–5]. On the other hand, if the network topology can somehow be altered, then additional flexibility is available, and the optimization can be performed over the topology as well as the weights [6–10]. Generally speaking, topology optimization is a very difficult combinatorial problem and different suboptimal approaches can be adopted. In [6] the convergence properties of different topology classes are theoretically analyzed on average, given the number of nodes of a general network. In [8] it is shown that, starting from a given topology, removing certain links can be beneficial in terms of convergence speed; this approach was later refined in [9]

☆ Research supported by the European Regional Development Fund (ERDF) and the Spanish Government (TEC2013-47020-C2-1-R COMPASS, CONSOLIDER-INGENIO 2010 CSD2008-00010 COMONSENS, TACTICA), and the Galician Regional Government (R2014/037, GRC2013/009 and AtlantTIC).

* Corresponding author. Tel.: +39 0461 408400; fax: +39 0461 421157.
E-mail addresses: massimo.vecchio@create-net.org (M. Vecchio), valcarce@gts.uvigo.es (R. López-Valcarce).

http://dx.doi.org/10.1016/j.ipl.2014.11.009
0020-0190/

in order to judiciously remove and add links with the goal of speeding up the consensus algorithm while keeping energy consumption at bay. In WSNs with static nodes, topology control can be achieved by varying the transmit power, as considered in [7,10].

All of these approaches implicitly assume *unicast* pairwise communication[1]: each node can independently set the transmit power it allocates to communicate with each of its neighbors, *e.g.*, by using orthogonal signaling. However, in distributed consensus schemes the information that nodes need to send at a given iteration is the same for all of its neighbors, so that in WSNs it is possible to exploit the broadcast nature of the wireless channel, also known as Wireless Multicast Advantage (WMA) [11]: at each consensus iteration, each node may broadcast its state while its neighbors simultaneously listen, thus reducing the number of required transmissions. On the other hand, exploiting WMA while varying the transmit power of a given node affects the links to *all* of its neighbors, so that these cannot be independently controlled now.[2] This motivates specific topology optimization strategies that take this fact into account, since previous topology control schemes designed under the unicast assumption cannot be applied under these "broadcast communication" constraints.

This problem is related to the so-called *range assignment* (RA) problem in broadcast WSNs, usually oriented to other network-related goals (*e.g.* maintaining global connectivity [13]) and known to be difficult in general [14]. Our goal is to determine the transmit power for each node in a broadcast WSN in order to minimize the convergence time of a given distributed average consensus scheme. One issue featuring in such setting is the fact that, if the transmit powers of nodes i and j are different (non-homogeneous RA [13]), it may well happen that node i is out of the coverage range of node j whereas node j can listen to node i's transmissions; in other words, the underlying graph becomes *directed*. This has implications for consensus algorithms. Although reaching an agreement over a directed graph is easily achieved, the agreement value will be a *weighted* average of the agents' states, and the weights will depend on the topology. When the *unweighted* average is of interest, certain stringent requirements on the directed graph must be imposed (such as some sort of graph balancing [15]), which are generally difficult to enforce in practice. Hence, we focus on undirected graphs, for which reaching an agreement on the *unweighted* average by consensus algorithms is not a problem. To this end, we adopt a simple strategy by which nodes just ignore transmissions received from neighbors which are not within their own transmit range (in the previous example, node j would simply ignore packets received from node i), thus obtaining an undirected topology. With this framework, we start from a maximally connected setting (all nodes transmit at full power), and then proceed to iteratively reduce the power of one node at a time in a centralized greedy fashion in order to maximize the convergence rate.

As previous approaches to topology control [6–10], ours is a centralized scheme which can be run by a central entity after deployment and previously to the network becoming operative; after such step, network operation may become decentralized. Fully distributed topology control methods are desirable and should be the target of future research.

In Section 2 the network model and the basics of consensus schemes are presented. The proposed greedy algorithm for non-homogeneous RA is presented in Section 3. Simulation results and conclusions are provided in Sections 4 and 5.

2. Problem setting

2.1. Graph model

Consider a set $\mathcal{V}$ of randomly deployed nodes with indices $i \in \{1, \cdots, n\}$. Let d_{ij} be the distance between nodes i and j, and let $\mathcal{R} = \{r_i \in [0, r_{\max}], i = 1, \cdots, n\}$ be a set of connectivity radii (*i.e.* transmit ranges), with $r_{\max} > 0$ the maximum allowable range. We adopt a simple model by which a link between two nodes exists iff their distance does not exceed the transmit range of the transmitter, which can be controlled by setting the transmit power [10, 13]. As discussed in Section 1, the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is defined as

$$(i, j) \in \mathcal{E} \quad \Leftrightarrow \quad i \neq j, \quad d_{ij} \leq \min\{r_i, r_j\}. \tag{1}$$

In this way, the graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is undirected as desired. If $r_i = r$ for $i = 1, \cdots, n$ we recover the homogeneous RA over the standard Random Geometric Graph (RGG) model [16]. However, we allow for different transmit ranges at different nodes in order to add flexibility to the design.

The neighborhood of node i is defined as $\mathcal{N}_i = \{j : (i, j) \in \mathcal{E}\}$, and its degree (number of neighbors) is therefore $|\mathcal{N}_i|$. The graph Laplacian matrix $\boldsymbol{L}$ has elements

$$\boldsymbol{L}_{ij} = \begin{cases} |\mathcal{N}_i| & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } j \in \mathcal{N}_i \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

By construction $\boldsymbol{L}$ is symmetric. Let $\lambda_1(\boldsymbol{L}) \leq \lambda_2(\boldsymbol{L}) \leq \cdots \leq \lambda_n(\boldsymbol{L})$ denote its ordered eigenvalues. Note that $\lambda_1(\boldsymbol{L}) = 0$ with corresponding eigenvector the $n \times 1$ all-ones vector $\mathbf{1}$. Moreover, $\lambda_2(\boldsymbol{L}) > 0$ iff $\mathcal{G}$ is connected [16]. $\lambda_2(\boldsymbol{L})$ is known as the *algebraic connectivity* of the graph.

2.2. Average consensus algorithms

Let $\boldsymbol{x}(0) = [x_1(0) \cdots x_n(0)]^T \in \mathbb{R}^n$ denote the vector of initial node measurements. The goal of the average consensus algorithm is to have all nodes compute the average $\bar{x} = \frac{1}{n}\mathbf{1}^T\boldsymbol{x}(0)$, iteratively and in a distributed fashion (thus node i can only communicate with nodes in $\mathcal{N}_i$). Distributed linear iterations [3] take the form $\boldsymbol{x}(k) = \boldsymbol{W}\boldsymbol{x}(k-1)$, where

[1] As an exception, in [10, Sec. V] a broadcast scheme is considered, but the transmit power is constrained to be equal for all nodes in the network, in contrast with the approach proposed in the present work.

[2] Transmissions in a broadcast WSN should be coordinated at the MAC layer in order to avoid collisions and align the listening and transmitting nodes, for instance by implementing some suitable time-synchronization protocol [12].

$\boldsymbol{W}$ is a symmetric weight matrix with zero (i,j) entries if $i \neq j$ and $(i,j) \notin \mathcal{E}$, and such that $\boldsymbol{W}\mathbf{1} = \mathbf{1}$ and $\rho(\boldsymbol{W} - \frac{1}{n}\mathbf{1}\mathbf{1}^T) < 1$, with $\rho(\cdot)$ denoting the spectral radius. Under these conditions, $\lim_{k\to\infty} \boldsymbol{x}(k) = \bar{x}\mathbf{1}$ as desired. The *asymptotic convergence factor* η and *convergence time* τ are defined as

$$\eta = \sup_{\boldsymbol{x}(0)\neq\bar{x}\mathbf{1}} \lim_{k\to\infty} \left(\frac{\|\boldsymbol{x}(k) - \bar{x}\mathbf{1}\|_2}{\|\boldsymbol{x}(0) - \bar{x}\mathbf{1}\|_2} \right)^{\frac{1}{k}}, \quad \tau = \frac{s}{\log(1/\eta)}. \tag{3}$$

The latter is indicative of the number of iterations required for the error norm $\|\boldsymbol{x}(k) - \bar{x}\mathbf{1}\|_2$ to decrease by a factor of e^{-s}. As shown in [3], one has $\eta = \rho(\boldsymbol{W} - \frac{1}{n}\mathbf{1}\mathbf{1}^T)$. Noting that $\boldsymbol{W}$ depends on the topology, the problem is then stated as

$$\min_{\mathcal{R}} \rho\left(\boldsymbol{W} - \frac{1}{n}\mathbf{1}\mathbf{1}^T\right). \tag{4}$$

Note that choosing $\mathcal{R}$ sets the zero-valued entries of $\boldsymbol{W}$. The nonzero entries are then set by some weight selection method, such as FDLA [3], max-degree, Metropolis [17], etc. Although the scheme proposed in Section 3 can be applied to any weight selection method, for illustrating purposes we focus on constant-weight assignments [1], in which $\boldsymbol{W} = \boldsymbol{I} - \alpha\boldsymbol{L}$ with $0 < \alpha < 2/\lambda_n(\boldsymbol{L})$ a stepsize. Given $\mathcal{R}$ and the corresponding topology with Laplacian $\boldsymbol{L}$, the optimum stepsize minimizing the asymptotic convergence factor η is

$$\alpha_\star = \frac{2}{\lambda_n(\boldsymbol{L}) + \lambda_2(\boldsymbol{L})} \quad \Rightarrow \quad \eta = \frac{1-\gamma(\boldsymbol{L})}{1+\gamma(\boldsymbol{L})}, \tag{5}$$

with $\gamma(\boldsymbol{L}) = \lambda_2(\boldsymbol{L})/\lambda_n(\boldsymbol{L})$.

The nonconvex problem (4) is quite hard, as most RA problems [14]. Next we present greedy approaches to solving (4) with a good tradeoff between complexity and performance.

3. A greedy approach to fast consensus

An important observation is that the *maximally connected* topology[3] obtained by setting $r_i = r_{\max}$ for all i is not necessarily optimal in terms of η. For example, with a constant weight assignment with optimum stepsize, minimizing η amounts to maximizing the eigenvalue ratio $\gamma = \lambda_2/\lambda_n$. It is known that the eigenvalues of the Laplacian matrix cannot increase (resp. decrease) by removing links from (resp. adding links to) a given topology [9,18]. If some of the r_i are decreased and some links are removed as a consequence, the value of γ may actually increase if λ_n decreases faster than λ_2 [8,9]. Note also that this effect is due to the fact that the optimum stepsize is topology dependent. Should one keep the stepsize α constant, the asymptotic convergence factor would be $\eta = 1 - \alpha\lambda_2(\boldsymbol{L})$ which can only increase if links are removed.

Our approach starts with the maximally connected topology, and then successively removes one link at a time in order to obtain the best possible value of η at each step.

[3] Note that the maximally connected topology is not necessarily fully connected.

Table 1
Greedy algorithm for best topology search.

1. Set $m = 0$. Let $\mathcal{R}_0 = \{r_i = r_{\max},\ i = 1, \cdots, n\}$.
2. At step m, and for $j = 1, \ldots, n$,
   (a) Let $\mathcal{G}_m^j = \{\mathcal{V}, \mathcal{E}_m^j\}$ be the graph associated to $\mathcal{R}_m^{j-}$.
   (b) Let $\boldsymbol{W}_m^j$ be the corresponding weight matrix, and
   $$\eta_{m,j} = \rho\left(\boldsymbol{W}_m^j - \frac{1}{n}\mathbf{1}\mathbf{1}^T\right).$$
3. Set $\mathcal{R}_{m+1} = \mathcal{R}_m^{j(m)-}$, where $j(m) = \arg\max_j \eta_{m,j}$.
4. Let $\mathcal{G}_{m+1} = \{\mathcal{V}, \mathcal{E}_{m+1}\}$ be the corresponding graph. If $\mathcal{G}_{m+1}$ is connected, and $m < m_{\max}$, set $m \leftarrow m+1$ and go to step 2.
5. Set $\mathcal{R} = \mathcal{R}_{m_\star}$, where $m_\star = \arg\max_m \eta_{m,j(m)}$.

In order to rigorously describe this scheme, let us first define the following two operations on a set of transmit ranges $\mathcal{R}$. If $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is the undirected graph corresponding to a given $\mathcal{R} = \{r_i\}$ as per (1), we define the *reduction* of $\mathcal{R}$ as

$$\mathcal{R}^- = \left\{r_i^-,\ i = 1, \cdots, n\right\}, \quad r_i^- = \max_{j|(i,j)\in\mathcal{E}} \{d_{ij}\}. \tag{6}$$

In words, each transmit range in $\mathcal{R}$ is decreased as much as possible without changing the structure of $\mathcal{G}$, by setting its value to the distance from the corresponding node to its farthest neighbor in $\mathcal{G}$. Clearly, a reduction always entails energy savings without affecting the topology; whereas any further decrease of any of the elements of $\mathcal{R}^-$ will change the underlying graph $\mathcal{G}$. We define the j-th *perturbation* of $\mathcal{R}^-$, denoted by $\mathcal{R}^{j-}$, as the set of transmit ranges obtained by (i) replacing r_j^- by $r_j^- - \epsilon$, where ϵ is a sufficiently small positive constant such that only the link between node j and its farthest neighbor is removed by this change, and (ii) applying a reduction to the resulting configuration.

The proposed iterative topology search is described in Table 1. At each step, the value of the objective function η obtained by removing the link between node j and its farthest neighbor is computed for every j, and the best choice is selected for the next step.[4] This is repeated for a maximum number of steps ($m_{\max}$ in step (4) of Table 1) or until a disconnected graph is obtained; the final configuration is selected from the history of topologies obtained along the process. Note that the number of links is reduced at each step, and thus each configuration will be sparser (and more energy-efficient) than that at the previous step.

The computational cost of the proposed scheme is dominated by step 2.b, which typically requires an eigenvalue decomposition (EVD) to compute the spectral radius.[5] Thus the total number of EVDs is $n \cdot m_{\max}$. This load can be reduced if we replace steps (2.b) and (3) in Table 1 by

(2.b′) Let $\delta_{m,j} = |\mathcal{N}_j|$, *i.e.*, the degree of node j at step m.

(3′) Set $\mathcal{R}_{m+1} = \mathcal{R}_m^{j(m)-}$, where $j(m) = \arg\max_j \delta_{m,j}$.

[4] A random choice can be made in case of ties.

[5] The fact that the Laplacian obtained after removing an edge is a rank-1 perturbation of the original Laplacian can be exploited in order to efficiently compute the EVD of the perturbed Laplacian from that of the original one; see [19, Sec. 8.5.3].

In this way, at each step the node with largest degree is chosen (also in this case a random choice is made in case of ties), and the link to its farthest neighbor is removed; the number of EVDs is then just $m_{\max}$. The rationale for this simplified scheme is that convergence speed tends to be larger in networks with more uniform degree distributions [6]. The following argument lends additional justification to this approach:

If the edge connecting vertices i and j is removed from the graph with Laplacian $\boldsymbol{L}$, the new Laplacian becomes $\tilde{\boldsymbol{L}} = \boldsymbol{L} - \boldsymbol{g}\boldsymbol{g}^T$, where $\boldsymbol{g} = \boldsymbol{e}_i - \boldsymbol{e}_j$, and $\boldsymbol{e}_k$ is the k-th column of the $n \times n$ identity matrix. Thus, $\tilde{\boldsymbol{L}}$ is a rank-1 perturbation of $\boldsymbol{L}$. Since $\|\boldsymbol{g}\|^2 = 2$, direct application of Theorem 8.1.8 from [19] yields the following upper bound on the eigenvalue ratio of the perturbed Laplacian:

$$\gamma(\tilde{\boldsymbol{L}}) \le \frac{\gamma(\boldsymbol{L})}{1 - \frac{2}{\lambda_n(\boldsymbol{L})}}. \tag{7}$$

Recall now [20] that $\lambda_n(\boldsymbol{L}) \ge \delta_{\max} + 1$, where $\delta_{\max} = \max_{1 \le i \le n} |\mathcal{N}_i|$ is the largest node degree of the graph. Combining this with (7) yields

$$\frac{\gamma(\tilde{\boldsymbol{L}})}{\gamma(\boldsymbol{L})} \le \frac{\delta_{\max} + 1}{\delta_{\max} - 1}. \tag{8}$$

Since the right-hand side of (8) is monotonically decreasing in $\delta_{\max}$, this suggests that the link to be removed be an edge corresponding to a node with largest degree.

As an example, consider $n = 50$ nodes randomly deployed in $[0, 1] \times [0, 1]$ with a constant-weight assignment. Fig. 1 shows the values of $\gamma = \lambda_2/\lambda_n$ obtained by the proposed greedy schemes as links get removed, for two different choices of $r_{\max}$. It is seen that the larger $r_{\max}$, the more links can be removed in order to optimize performance, which makes sense. The simplified greedy search seems to provide a good low-cost alternative to the full greedy scheme. As regards the execution times of the proposed optimization techniques, the simplified greedy approach (resp., the full-greedy approach) required 0.30 and 0.67 seconds (resp. 6.01 and 11.67 seconds) to run to completion (*i.e.*, until disconnection) on the considered network topologies, respectively.[6]

Nevertheless, it should be pointed out that the simplified greedy technique should be cautiously used as an alternative to the full version, especially when trying to optimize peculiar network topologies. For example, if the initial graph is regular (*i.e.*, all nodes have the same degree), the simplified greedy version would simply start removing the farthest link of a randomly chosen node, and this would be repeated successively over the set of remaining nodes (again, randomly choosing the node among the remaining ones with the largest degree at each iteration). An even more peculiar situation in which the simplified greedy version would fail to select a convenient sub-optimal link to remove is the following: let assume

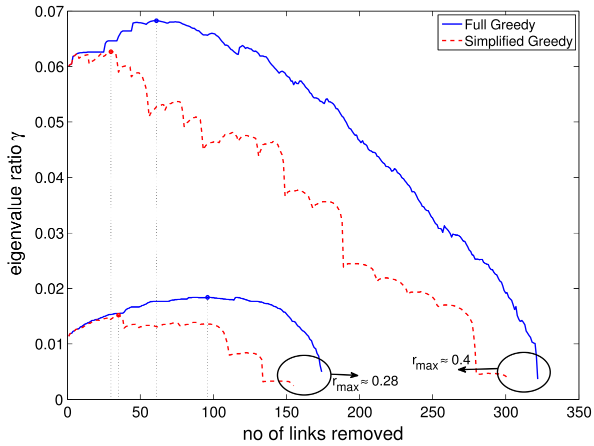


Fig. 1. γ versus number of steps of the proposed greedy methods for an example with a WSN with $n = 50$ nodes. The best γ values are marked with dots.

that node i has the largest degree and its farthest neighbor j has a node degree equal to 1 (*e.g.*, it can only communicate with node i). By design, the simplified greedy approach would remove link (i, j), hence causing an early network disconnection.

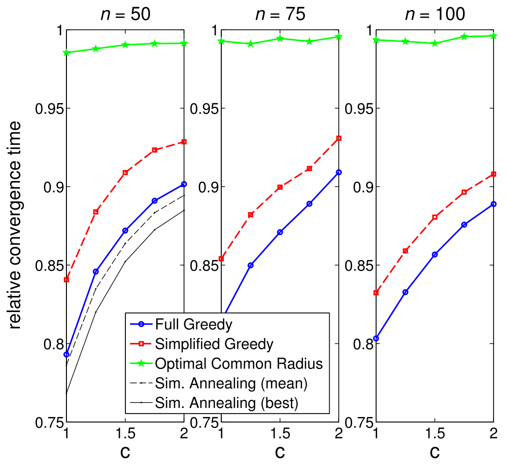
The trend for γ along the iterations of the greedy schemes, illustrated in Fig. 1, is generally observed: an increasing stage up to the maximum value followed by a decreasing stage until the network eventually becomes disconnected (and thus $\gamma = 0$) as a consequence of removing too many links. This general trend, however, is peppered with rather unpredictable local maxima and minima, so that one must in general complete the link removal process until network disconnection, and then look back in order to pick the best setting encountered in previous iterations.

4. Simulation results

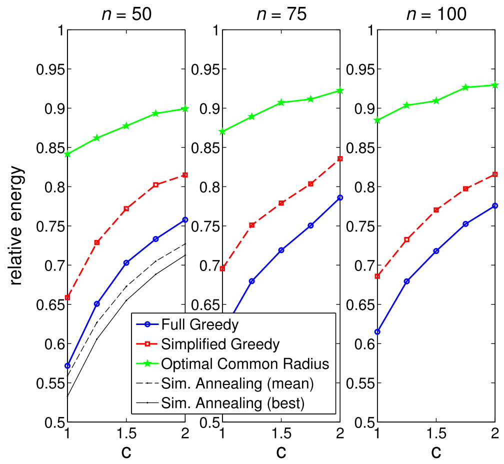
The performance of the greedy schemes was checked in a number of randomly generated deployments in $[0, 1] \times [0, 1]$. The maximum transmit range is obtained as a function of the number of nodes n as $r_{\max} = \sqrt{c \frac{\log n}{n}}$; thus, the larger the value of c, the more connected the topology. For $n = 50, 75$ and 100 nodes and $1 \le c \le 2$, we averaged the results over 100 random deployments for each (n, c) pair. A constant-weight assignment $\boldsymbol{W} = \boldsymbol{I} - \alpha_\star \boldsymbol{L}$ is assumed.

Fig. 2(a) shows the relative convergence time τ/τ_{mc}, where τ_{mc} denotes the convergence time corresponding to the maximally connected configuration (in the sequel we take $s = 7$ in (3), which amounts to a decrease of the error norm to 0.1% of its initial value). It is seen that it is possible to bring down the relative convergence time to the range 80%–90%, with a larger payoff for less connected settings. For comparison, we also show the results obtained with a strategy that constrains the transmit ranges to be equal at all nodes. Under this constraint, the optimum setting can be obtained by solving a scalar optimization problem in the variable $r \in [0, r_{\max}]$. Although this approach provides some advantage with respect to

[6] Our simulation framework was entirely developed in Matlab 8.1.0.604 (R2013a) for a 32-bit Windows 7 OS, running on a standard laptop machine equipped with 2 Intel(R) Core(TM) i7-2620M CPUs @ 2.70 GHz and 4 GB of RAM.



(a) convergence time.



(b) energy consumption.

Fig. 2. Relative (w.r.t. the maximally connected topology) performances of the different designs. Recall that the smaller the relative value, the better the corresponding metric of the sparsified configuration w.r.t. the maximally connected topology.

the maximally connected setting, the proposed methods yield much larger improvements. Thus, the benefit of allowing for different transmit ranges at different nodes is clear.

Although our goal is convergence speed, it is also worth checking the energy savings obtained by topology sparsification. Given a configuration $\mathcal{R} = \{r_i\}$, the average energy consumption after a consensus round is proportional to $\tau \cdot \frac{1}{n}\sum_{i=1}^{n} r_i^{\beta}$, with β the path-loss exponent and τ the convergence time as in (3). Thus, energy savings are due to a reduction in convergence time as well as in transmit power.[7] Fig. 2(b) shows the average energy consumption (also relative to that of the maximally connected setting), for $\beta = 2$. Savings in the range of 60%–80% are observed; again, less connected configurations seem to benefit most from sparsification.

The proposed schemes are greedy search methods with no guarantee of optimality. Thus, the question of how much could be gained in terms of performance with respect to the greedy solutions is pertinent. We ran a simulated annealing (SA)-based metaheuristic solver [21] which seeks the configuration with smallest τ starting from the maximally connected setting. We adopt a standard cooling law $t_k = 0.94 t_{k-1}$ with initial temperature $t_0 = 0.1$, and a standard Metropolis–Hastings acceptance probability function, while at t_0 hill climbings are accepted with probability 0.8 (disconnected configurations are always discarded). At each t_k value, each transmit range is randomly changed 15 times. The best setting found after a maximum of 5×10^4 EVDs is then returned. 10 independent SA runs were executed per deployment; for complexity reasons, only networks with $n = 50$ nodes were considered. The results in terms of relative convergence time and relative energy are shown in Figs. 2(a) and 2(b) respectively, both when picking the best result out of the 10 trials per setting and when averaging those 10 results. Although the greedy methods show some loss, they provide a good performance/complexity tradeoff, as their computational cost is much less than that of metaheuristic solvers such as SA.

5. Conclusions

New methods to optimize a topology-dependent cost function in the context of broadcast WSNs have been introduced. They start with a dense topology and successively remove one link at a time in a greedy fashion; the best configuration among those obtained is then picked. The methods effectively improve convergence speed for average consensus algorithms, with reduced energy consumption as an important side benefit. The full greedy version requires n EVDs of an $n \times n$ matrix per step, with n the network size, whereas the simplified version requires just one such EVD per step. Thus, with very large networks, the tradeoff between complexity and performance provided by the simplified greedy approach becomes appealing.

Certain assumptions underlying the model, *e.g.* knowledge of the node locations, circular transmission coverage regions, and a continuous range of feasible transmit powers, may not hold in practice. One may just have, *e.g.*, Signal-to-Noise Ratio (SNR) estimates for the links between pairs of nodes. In that case, the inverse SNR can be used as a proxy for distance, and then the true shape of coverage regions becomes irrelevant. The greedy search methods can then be applied if one assumes that links are established only if the corresponding SNR is above a threshold guaranteeing successful message decoding: the role of the "farthest neighbor" corresponds to the neighboring node with worst SNR still above the threshold. If the corresponding link is to be dropped, then the transmit power is reduced by the difference (in dB) between the SNR value and the threshold. Discrete values of transmit power, as is often the case in practical chipsets, are straightforwardly accommodated as well. Future work should address the impact of these practical issues as well as others such as node and/or link failures in the network.

[7] We only consider *transmit* power in the energy budget, since receive energy consumption ultimately depends on the type of radio and hardware implementation.

References

[1] R. Olfati-Saber, J.A. Fax, R.M. Murray, Consensus and cooperation in networked multi-agent systems, Proc. IEEE 95 (1) (2007) 215–233.

[2] A.G. Dimakis, S. Kar, J.M.F. Moura, M.G. Rabbat, A. Scaglione, Gossip algorithms for distributed signal processing, Proc. IEEE 98 (11) (2010) 1847–1864.

[3] L. Xiao, S. Boyd, Fast linear iterations for distributed averaging, Syst. Control Lett. 53 (2004) 65–78.

[4] L. Xiao, S. Boyd, S.J. Kim, Distributed average consensus with least-mean-square deviation, J. Parallel Distrib. Comput. 67 (1) (2007) 33–46.

[5] K. Avrachenkov, M. El Chamie, G. Neglia, A local average consensus algorithm for wireless sensor networks, in: Proc. DCOSS, 2011, pp. 1–6.

[6] S. Kar, J.M.F. Moura, Topology for global average consensus, in: Proc. ASILOMAR, 2006, pp. 276–280.

[7] S. Barbarossa, G. Scutari, A. Swami, Achieving consensus in self-organizing wireless sensor networks: the impact of network topology on energy consumption, in: Proc. ICASSP, 2007, pp. 841–844.

[8] C. Asensio-Marco, B. Beferull-Lozano, Accelerating consensus gossip algorithms: sparsifying networks can be good for you, in: Proc. ICC, 2010, pp. 1–5.

[9] C. Asensio-Marco, B. Beferull-Lozano, A greedy perturbation approach to accelerating consensus algorithms and reducing its power consumption, in: Proc. SSP, 2011, pp. 365–368.

[10] S. Sardellitti, S. Barbarossa, A. Swami, Optimal topology control and power allocation for minimum energy consumption in consensus networks, IEEE Trans. Signal Process. 60 (1) (2012) 383–399.

[11] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, On the construction of energy-efficient broadcast and multicast trees in wireless networks, in: Proc. INFOCOM, vol. 2, 2000, pp. 585–594.

[12] B. Sundararaman, U. Buy, A.D. Kshemkalyani, Clock synchronization for wireless sensor networks: a survey, Ad Hoc Netw. 3 (2005) 281–323.

[13] P. Santi, Topology control in wireless ad hoc and sensor networks, ACM Comput. Surv. 37 (2) (2005) 164–194.

[14] B. Fuchs, On the hardness of range assignment problems, Networks 52 (4) (2008) 183–195.

[15] R. Olfati-Saber, R.M. Murray, Consensus problems in networks of agents with switching topology and time-delays, IEEE Trans. Autom. Control 49 (9) (2004) 1520–1533.

[16] C. Godsil, G. Royle, Algebraic Graph Theory, Graduate Texts in Mathematics, vol. 207, Springer, 2001.

[17] L. Xiao, S. Boyd, S. Lall, A scheme for robust distributed sensor fusion based on average consensus, in: Proc. ICIPSN, 2005, pp. 63–70.

[18] M. Fiedler, Algebraic connectivity of graphs, Czechoslov. Math. J. 23 (1973) 298–305.

[19] G.H. Golub, C.F. Van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore, MD, 1996.

[20] R. Grone, R. Merris, The Laplacian spectrum of a graph II, SIAM J. Discrete Math. 7 (2) (1994) 221–229.

[21] M. Gendreau, J.Y. Potvin (Eds.), Handbook of Metaheuristics, 2nd edition, International Series in Operations Research & Management Science, Springer Publishing Company, Incorporated, 2010.