Sergio Barbarossa and Gesualdo Scutari

Bio-Inspired Sensor Network Design

Distributed decisions through self-synchronization

The goal of this article is to show how a simple self-synchronization mechanism, borrowed from biological systems, can form the basic tool for achieving globally optimal distributed decisions in a wireless sensor network with no need for a fusion center. After describing the basic interaction mechanism among the network nodes, we will illustrate the conditions guaranteeing the convergence of each node to a global (or local) consensus coinciding with the globally (locally) optimal decision statistics. The interaction mechanism takes into account the physical channel parameters, such as fading coefficients and propagation delays. We then illustrate our results through examples of distributed estimation and multiple hypothesis testing. We also address energy consumption issues and discuss some possible implementations.

1053-5888/07/$25.00©2007IEEE

OVERVIEW

The design of sensor networks faces a number of challenges resulting from very demanding requirements on one side, such as high reliability of the decision taken by the network and robustness to node failure, and very limited resources on the other side, such as energy, bandwidth, and node complexity. For this reason, many recent works on sensor networks have concentrated on the efficient use of the available resources, mainly energy, necessary to achieve the users' requirements [1]. Given the considerable amount of knowledge accumulated in the telecommunication networks field, it is not surprising that most works on sensor networks fall within the conceptual framework of telecommunication networks, with a special emphasis on energy-efficient design, node simplicity, and scalability. However, while the goal of a telecommunication network is to carry information packets from each source to the relative destination, *irrespective* of the packet content, a sensor network is, fundamentally, an *event-driven* system: in a sensor network, it is events that are carried and not packets. If a fire starts, what is important is that the remote control or actuator node gets this information as soon as possible, not that all the temperature measurements taken by the sensors reach the fusion center.

A critical aspect of a sensor network is its vulnerability to temporary node sleeping, due to duty-cycling for battery recharge, permanent failures, or even intentional attacks. Clearly, the vulnerability increases if there are just a few decision (sink) nodes, as the failure of a sink node could jeopardize the whole system. To improve resilience against node failures, it is necessary to devise decentralized decision strategies that are able to quickly react to unpredictable topology changes.

Decentralizing the decisions is also strategic to reducing the congestion probability. A congestion around a sink node is an event that is most likely to occur precisely when a hazard situation occurs, in which case many nodes send their warning packets to the control nodes at about the same time. This could make the network less reliable just when the monitoring system is expected to be as reliable as possible. Decentralizing the decisions is useful also in this case, as it would prevent the bottleneck situation in which all the nodes try to access a single sink node.

Of course, the network resilience to node failures increases as the number of nodes increases. However, this raises the *scalability* issue. From an information theoretic point of view, it was shown in [2] that the transport capacity of a sensor network composed of N nodes sending data to a sink node scales as $1/N$. While intuitive, this result is a somewhat discouraging, as it implies that we cannot do any better than just dividing the resources (e.g., bandwidth) by the number of nodes. Nevertheless, as shown by Giridhar and Kumar [3], [4] this is not necessarily the case if the specificity of the sensing problem is properly taken into account. A sensor network can be in fact seen as a sort of distributed computer that, on the basis of the data collected by N sensors, let us say $y_1, y_2, \ldots, y_N$, has to compute a function $f(y_1, y_2, \ldots, y_N)$. This function may be, for example, a *sufficient statistic* of the data. From this perspective, a sensor network could be designed using the powerful tools of parallel and distributed optimization [5], [6]. Moreover, the objective function $f(y_1, y_2, ldots, y_N)$ may have properties that can be exploited to improve scalability. In many applications, for example, $f(y_1, y_2, \ldots, y_N)$ is a *symmetric* function of the measurements; i.e., it is invariant to any permutation of the observed variables. In such a case, it was shown in [4] that the transport capacity scales as $1/\log N$, as opposed to $1/N$. Interestingly, this permutation invariance assumption is not at all artificial, as, conversely, it reflects the *data-centric* nature of some sensor networks, where what is important is the whole set of measurements and not the knowledge of which node has taken which measurement. Examples of networks designed according to the data-centric perspective include the *type-based* multiple access, where the identification of the data type at the sensor decides what orthogonal channel the sensor is going to use over a multiple access channel [7]. A sensor network can be seen indeed as a *multiterminal inference machine*, where multiple nodes send their information to a control node, subject to a rate constraint imposed by the communication channel [1]. This induces an interplay between signal processing and networking that should be properly taken into account [1].

Besides deriving scaling laws pertinent to sensor networks, Giridhar and Kumar [4] showed also that better scalability laws can be achieved by endowing the network with *in-network* processing capabilities and structuring the network in a hierarchical structure, with different roles assigned to different nodes.

Following [4], we share the view that the network should be organized in hierarchical levels, where the lower level nodes, simple but vulnerable, cooperate to achieve local consensus with a reliability greater than the single node, whereas intermediate nodes are responsible for conveying the local consensus achieved by the lower level nodes to the control centers. In this article, we concentrate on the lower level nodes and our goal is to illustrate a strategy of interaction among the nodes that allows them to reach globally optimal decisions with a totally decentralized approach, exploiting a consensus mechanism based on the self-synchronization of first-order coupled dynamical systems. The basic idea is borrowed from biological systems where self-synchronization occurs in a variety of circumstances, from the heart beating to neuron cell firing, and forms the basis of their robustness and reliability. Our goal is to bring the mathematical models describing self-synchronization into a signal processing perspective to derive a novel approach to distributed hypothesis testing or estimation.

Distributed consensus algorithms are indeed techniques largely studied in distributed computing (see, e.g., [6]) and their application to statistical consensus theory has a long history (see, e.g., [8]). Average consensus techniques have received great attention in recent years (see, e.g., [9]–[13] and references therein). In particular, the conditions for achieving a consensus over a common specified value, like a linear combination of the observations, was solved for networked dynamic systems by Olfati-Saber and Murray, under a variety of network topologies, also allowing for topology variations during the time necessary

to achieve consensus [9], [10], [13]. Belief propagation is another powerful distributed technique capable of achieving globally optimal decisions in a totally decentralized way (see, e.g., [14]–[16] and the references therein). A special form of consensus, known as the *alignment* problem, where all network agents eventually reach an agreement, but without specifying the final value, was also thoroughly studied by Jadbabaie et al. in [17]. A recent excellent tutorial on distributed consensus techniques is given in [13].

MOTIVATING EXAMPLES

In this section we will illustrate some basic statistical signal processing problems, whose centralized solution is well known. The goal of the ensuing sections will be to show how to solve them in a totally decentralized way.

PROBLEM 1: ESTIMATION OF A COMMON SET OF PARAMETERS

Let us consider a network composed of N sensors, whose goal is to estimate a set of L parameters, given by vector ξ. We assume that each sensor collects a vector $\mathbf{y}_i$ of M measurements, related to the unknown vector ξ through the linear observation model

$$\mathbf{y}_i = \mathbf{A}_i\, \xi + \mathbf{v}_i\,; \tag{1}$$

where $\mathbf{v}_i$ is additive noise, with zero mean and covariance matrix $\mathbf{R}_i$. The noises on different sensors are statistically independent of each other but each vector $\mathbf{v}_i$ may be colored. If the noise probability density function (pdf) is Gaussian, the globally optimal maximum likelihood (ML) estimator is [18]

$$\hat{\xi} = \left(\sum_{\mathrm{i}=1}^{\mathrm{N}} \mathbf{A}_i^T \mathbf{R}_i^{-1} \mathbf{A}_i\right)^{-1} \left(\sum_{\mathrm{i}=1}^{\mathrm{N}} \mathbf{A}_i^T \mathbf{R}_i^{-1} \mathbf{y}_i\right). \tag{2}$$

If the noise pdf is unknown, (2) still represents a meaningful estimator, as it is the best linear unbiased estimator (BLUE) [18]. If this estimation is to be taken by a fusion center, every sensor is required to transmit to the fusion center not only its measurement vector $\mathbf{y}_i$ but also its own mixing matrix $\mathbf{A}_i$ and noise covariance matrix $\mathbf{R}_i$.

PROBLEM 2: MULTIPLE HYPOTHESIS TESTING

Let us consider now the case where the goal of the network is to distinguish between M alternative hypotheses, with $M \geq 2$, in general. The M hypotheses may be spatial patterns, for example. We suppose that each hypothesis $\mathcal{H}_k$ occurs with a known a priori probability P_k and we denote by $p_i(\mathbf{y}_i/\mathcal{H}_k)$ the conditional pdf related to the observation of vector $\mathbf{y}_i$, conditioned to the hypothesis $\mathcal{H}_k$. We also assume that the measurements taken by different sensors are conditionally independent from each other so that, if we group the vector of measurements $\mathbf{y}_i$ into a vector $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N)^T$, we have $p(\mathbf{y}/\mathcal{H}_k) = \Pi_{i=1}^{N} p_i(\mathbf{y}_i/\mathcal{H}_k)$. If the decision criterion consists of minimizing the error probability, the optimal decision test is the one that chooses the hypothesis $\hat{\mathcal{H}}_m$ that maximizes the a posteriori probability $P(\mathcal{H}_k/\mathbf{y})$, or [18]

$$\begin{aligned} \hat{\mathcal{H}}_m &= \arg\max_k \{p(\mathbf{y}/\mathcal{H}_k) P(\mathcal{H}_k)\} \\ &= \arg\max_k \left\{ \prod_{i=1}^{N} p_i(\mathbf{y}_i/\mathcal{H}_k) P(\mathcal{H}_k) \right\}. \end{aligned} \tag{3}$$

Hence, the fusion center needs to know every single pdf $p_i(\mathbf{y}_i/\mathcal{H}_k)$ to take a decision.

DECENTRALIZED DECISION THROUGH SELF-SYNCHRONIZATION

In this section we will briefly review the self-synchronization concept, as observed in nature, and then we will show how to use a self-synchronization mechanism to force every single node of the network to reach the globally optimal decision tests specified in the previous section, without the need of any a fusion center.

A BRIEF HISTORY OF THE SELF-SYNCHRONIZATION IDEA

Self-synchronization is a phenomenon first observed between pendulum clocks (hooked to the same wooden beam) by Christian Huygens in 1658 [19]. Since then, self-synchronization has been observed in a myriad of natural phenomena, from flashing fireflies in South East Asia to singing crickets, from cardiac pacemaker or neuron cells to menstrual cycles of women living in strict contact with each other [19]. Readers interested in the self-synchronization phenomena as observed in nature should refer to [19] for an in-depth treatment of this fascinating and pervasive subject. A simple example may be useful to illustrate how some basic biological mechanisms could be exploited to devise robust sensor networks with built-in consensus capabilities. In the so-called pacemaker cells present in the human heart, a chemical reaction takes place that generates positive ions. This reaction induces an electrical potential difference, between the interior and exterior of the cell, that increases with time. Above a certain potential difference, however, the cell membrane becomes transparent so that the internal ions are fired to the exterior, thus periodically resetting the potential to zero. This behavior makes the single cell work as a pulse oscillator. At the same time, the cell membrane lets outer ions, fired by neighbor cells, enter into the cell. This induces an interaction about the firing times so that, from the outside, the overall population of pacemaker cells can be seen as a set of pulse-coupled oscillators. The situation is actually more complicated, as this population is also affected by pulsed commands arriving from the brain, which reacts to external and internal stimuli to insure the proper functioning of the whole body. From an engineering point of view, the main question is whether this system is sufficiently stable to guarantee a proper functioning, in spite of the simplicity and potential unreliability of the single cell. Interestingly, Mirollo and Strogatz provided a rigorous answer to this question proving that, under mild assumptions on the coupling function, if the network of nodes is fully connected, there exists only one stable equilibrium represented by the situation where all cells fire at the same time [20]. Hence, the interaction among the

cells is what insures the desired robustness. Actually, full connectivity is not even necessary, as what is really needed is only global connectivity (i.e., the presence of a path between every pair of cells). The previous mechanism is very robust, as it can tolerate the death of many cells without any permanent impact on the cell firing rate, provided that the number of dead cells does not affect the network connectivity. Hoppensteadt and Izhikevich went even further, proving that two pulse-coupled oscillators really interact with each only if their rates are in a resonant ratio. This led them to conjecture that simultaneous exchanges of information can take place in the human brain using a sort of frequency multiplexing: Classes of neurons tuned on the same (or resonant) firing rate interact with each other, but they do not interact with other classes of neurons firing at nonresonant rates [21].

The previous mechanism is just an example of an algorithm inspired by biological systems that could have an impact in the design of robust, self-organizing, systems. For example, Izhikevich proposed the use of a pulse-coupled oscillator as an associative memory [22]. As an example related to sensor networks, Hong et al. proposed the use of pulse-coupled oscillators to achieve a detection consensus in a fully connected sensor network [23], building on the results of Mirollo and Strogatz [20]. Later, Lucarelli et al. [24] proved that full connectivity is not necessary, as what is strictly required is global connectivity (i.e., the existence of a path between each pair of nodes, possibly composed of several hops). The approach suggested in [23], [24] is a form of consensus achieved through self-synchronization, but the final consensus value is a priori unpredictable. Also, timing synchronization, as proposed in [23], [24], may be critical in wide-area networks, where propagation delays might induce an ambiguity problem. In general, self-synchronization may be extended to include the situation where a consensus is achieved over a physical parameter of interest, not necessarily the initial time of phase. In this broader sense, self-synchronization was proposed as the way to achieve globally optimal estimation or detection [25]–[29]. In the next sections we will concentrate on these approaches.

OPTIMAL DISTRIBUTED DECISION THROUGH SELF-SYNCHRONIZATION

The idea of sensor networks proposed in [25]–[29] is the following. The network is composed of N nodes, each equipped with four basic components: 1) a *transducer* that senses the physical parameter of interest y_i (e.g., temperature, concentration of contaminants, radiation, etc.); 2) a *local processing unit* that provides a function $g_i(y_i)$ of the measurement; 3) a *dynamical system*, initialized with the local measurements, whose state $x_i(t)$ evolves as a function of its own measurement $g_i(y_i)$ and of the state of nearby sensors; 4) a *radio interface* that makes possible the interaction among the sensors. In the scalar observation case, the state of the dynamical system present at node i evolves according to the following differential equation [25]–[29]:

$$\dot{x}_i(t) = g_i(y_i) + \frac{K}{c_i}\sum_{j=1}^{N} a_{ij}\, h[x_j(t-\tau_{ij}) - x_i(t)] + w_i(t), \quad i = 1, 2, \ldots, N, \tag{4}$$

where $h(x)$ is a scalar coupling function, typically dependent on the radio interface (if $h(x) = \sin(x)$, (4) coincides with Kuramoto's model [30]), K is a global control loop gain, c_i is a local positive coefficient whose choice determines the final consensus form, the coefficients a_{ij} account for the coupling among the nodes, τ_{ij} is the propagation delay from node j to node i, and $w_i(t)$ is coupling noise. The only assumption we make on the coefficients a_{ij} is that they are non-negative. They may depend, for example, on the channel parameters according to the law $a_{ij} = \sqrt{p_j |h_{ij}|^2 / d_{ij}^2}$, where p_j is the power transmitted by node j, h_{ij} is the fading coefficient of the channel between nodes i and j, and d_{ij} is the distance between nodes i and j. In general, the coefficients a_{ij} are not symmetric, to accommodate, as an example, for different transmit powers. The evolution of the state equation (4), for example, for $t > 0$, requires the specification of the states $x_i(t), \forall i$, for a time interval $[-\tau_{\max}, 0]$, where $\tau_{\max}$ is the maximum propagation delay [for unretarded systems, it is only necessary to specify $x_i(0)$]. In the vector case, each node observes a vector $\mathbf{y}_i$ and the state equation becomes

$$\dot{\mathbf{x}}_i(t) = \mathbf{g}_i(\mathbf{y}_i) + K\, \mathbf{C}_i^{-1} \sum_{j=1}^{N} a_{ij}\, \mathbf{h}[\mathbf{x}_j(t-\tau_{ij}) - \mathbf{x}_i(t)] + \mathbf{w}_i(t), \quad i = 1, 2, \ldots, N, \tag{5}$$

where $\mathbf{C}_i$ is a positive definite matrix. The important question now is how to choose the free parameters of (4) or (5) and to determine the conditions guaranteeing that each node is able to reach the globally optimal decision statistic. But, before proceeding it is necessary to clarify what we mean by consensus. We say that the system (4) synchronizes if all the state *derivatives* $\dot{x}_i(t)$ converge asymptotically toward a common value ω^*, for any set of initial conditions. A global consensus is achieved when the whole system synchronizes in the sense specified above.

This definition is different from the commonly used definition in which what is required is convergence on the state (see, e.g., [12]), rather than on its derivative. The main motivation underlying our choice is the better resilience against coupling noise. In fact, in the presence of coupling noise [$w_i(t)$ in (4)], the states $x_i(t)$, at least in the simple linear coupling case, are affected by a random walk process whose variance increases linearly with time. This makes the final consensus a strong function of the coupling noise. In our case, instead, the state derivative is affected by a noise that is the derivative of a random walk and then it has a constant variance. Furthermore, as shown in [28], system (4) is able to converge, for *any* set of (bounded) propagation delays, to a prescribed function of the observations. Conversely, classical consensus techniques [10], [13], studied under a homogeneous delay assumption (i.e.,

$\tau_{ij} = \tau$), converge if the (undirected) graph is connected, and if and only if the delay is less than a topology-dependent value.

It is now useful to recall some basic properties of directed graphs, as they represent the basic analytical tool for studying the interaction among the nodes.

DIRECTED GRAPHS: THE BASIC MATHEMATICAL TOOL TO DESCRIBE INTERACTIONS

A weighted directed graph (or digraph, for short) $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is defined as a set of nodes (or vertices) $\mathcal{V}$ and a set of edges $\mathcal{E}$ (i.e., *ordered* pairs of nodes), with the convention that $e_{ij} \triangleq (v_i, v_j) \in \mathcal{E}$ (i.e., v_i and v_j are the head and the tail of the edge e_{ij}, respectively) means that the information flows from v_j to v_i. A digraph is weighted if a positive weight a_{ij} is associated to each edge. In our case, there are no loops, so that $a_{ii} = 0$. If $(v_i, v_j) \in \mathcal{E} \Leftrightarrow (v_j, v_i) \in \mathcal{E}$, then the graph is said to be *undirected*. For any node $v_i, \mathcal{N}_i$ is the set of neighbors of node v_i (i.e., the set of nodes sending data to node v_i). A *strong path* in a digraph $\mathcal{G}$ is a sequence of *distinct* nodes $v_1, v_2, \ldots, v_q \in \mathcal{V}$ such that $(v_{i-1}, v_i) \in \mathcal{E}$, for $i = 2, \ldots, q$. If $v_1 \equiv v_q$, the path is said to be *closed*. A *weak* path is a sequence of *distinct* nodes $v_1, v_2, \ldots, v_q \in \mathcal{V}$ such that either $(v_{i-1}, v_i) \in \mathcal{E}$ or $(v_i, v_{i-1}) \in \mathcal{E}$, for $i = 2, \ldots, q$. A strong *cycle* (or circuit) is a closed strong path. A digraph with N nodes is a *directed tree* if it has $N - 1$ edges and there exists a distinguished node, called the *root* node, which can reach *all* the other nodes through a (unique) strong path. Hence, a directed tree cannot have cycles and every node, except the root, has one and only one incoming edge. A digraph is a (directed) *forest* if it consists of one or more directed trees. A subgraph $\mathcal{G}_s = \{\mathcal{V}_s, \mathcal{E}_s\}$ of a digraph $\mathcal{G}$, with $\mathcal{V}_s \subseteq \mathcal{V}$ and $\mathcal{E}_s \subseteq \mathcal{E}$, is a directed *spanning* tree (or a *spanning* forest) if it is a directed tree (or a directed forest) and it has the *same* node set as $\mathcal{G}$; i.e., $\mathcal{V}_s \equiv \mathcal{V}$. We say that a digraph $\mathcal{G}$ contains a spanning tree (or a spanning forest) if there exists a subgraph of $\mathcal{G}$ that is a directed spanning tree (or a spanning forest).

In a digraph there are many degrees of connectedness: 1) a digraph is *strongly* connected (SC) if any ordered pair of distinct nodes can be joined by a *strong* path; 2) a digraph is *quasi strongly* connected (QSC) if, for every ordered pair of nodes v_i and v_j, there exists a node r that can reach either v_i or v_j via a *strong* path; 3) a digraph is *weakly* connected (WC) if any ordered pair of distinct nodes can be joined by a *weak* path; 4) a digraph is *disconnected* if it is not weakly connected. According to the above definitions, it is straightforward to see that strong connectivity implies quasi strong connectivity and that quasi strong connectivity implies weak connectivity, but the converse, in general, does not hold. For undirected graphs, instead, the above notions of connectivity are equivalent: An undirected graph is connected if any two distinct nodes can be joined by a path. Moreover, it is easy to check that the quasi strong connectivity of a digraph is equivalent to the existence of a directed spanning tree in the graph. When a digraph $\mathcal{G}$ is WC, it may still contain strongly connected subgraphs. A maximal subgraph of $\mathcal{G}$, which is also SC, is called a *strongly connected component* (SCC) of $\mathcal{G}$ [31]. Since a node is SC, it follows that every node lies in an SCC. Using this concept, any digraph $\mathcal{G}$ can be partitioned into SCCs, let us say $\mathcal{G}_1 = \{\mathcal{V}_1, \mathcal{E}_1\}, \ldots, \mathcal{G}_K = \{\mathcal{V}_K, \mathcal{E}_K\}$, where $\mathcal{V}_k \subseteq \mathcal{V}$ and $\mathcal{E}_k \subseteq \mathcal{E}, k = 1, \ldots, K$, denote the set of nodes and edges lying in the kth SCC, respectively.

The connectivity properties of a digraph may be better understood by referring to its corresponding *condensation* digraph. We may reduce the original digraph $\mathcal{G}$ to the condensation digraph $\mathcal{G}^* = \{\mathcal{V}^*, \mathcal{E}^*\}$ by associating the node set $\mathcal{V}_k$ of each SCC $\mathcal{G}_k$ of $\mathcal{G}$ to a single distinct node $v_k^* \in \mathcal{V}_k^*$ of $\mathcal{G}^*$ and introducing an edge in $\mathcal{G}^*$ from v_i^* to v_j^* if and only if there exists *some* edges from the SCC $\mathcal{G}_i$ and the SCC $\mathcal{G}_j$ of the original graph [31, Ch. 3.2]. An SCC that is reduced to the root of a directed spanning tree of the condensation digraph is called *root* SCC (RSCC). Looking at the condensation graph, we may single out the following topologies of the original graph: 1) *$\mathcal{G}$ is SC if and only if $\mathcal{G}^*$ is composed by a single node;* 2) *$\mathcal{G}$ is QSC if and only if $\mathcal{G}^*$ contains a directed spanning tree;* 3) *if $\mathcal{G}$ is WC, then $\mathcal{G}^*$ contains either a spanning tree or a (weakly) connected forest.*

Some examples of graph topologies are shown in the top row of Figure 1, where we report three topologies (top row), namely: (a) an SC digraph, (b) a QSC digraph with three SCCs, and (c) a WC (not QSC) digraph with a two-tree forest. For each digraph, we also sketch its decomposition into SCCs and RSCCs.

We recall now the basic properties of the matrices associated to a digraph, as they play a fundamental role in the stability analysis of the system illustrated in this article. Given a digraph $\mathcal{G}$, we introduce the following matrices associated with $\mathcal{G}$: 1) The $N \times N$ *adjacency matrix* $\mathbf{A}$, which is composed of entries a_{ij} equal to the weight associated to the edge e_{ij}, if $e_{ij} \in \mathcal{E}$, or equal to zero, otherwise; 2) the *degree matrix* $\mathbf{D}$, which is a diagonal matrix with diagonal entries $D_{ii} = \sum_{j=1}^{N} a_{ij}$; 3) the weighted *Laplacian* $\mathbf{L}$, defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$.

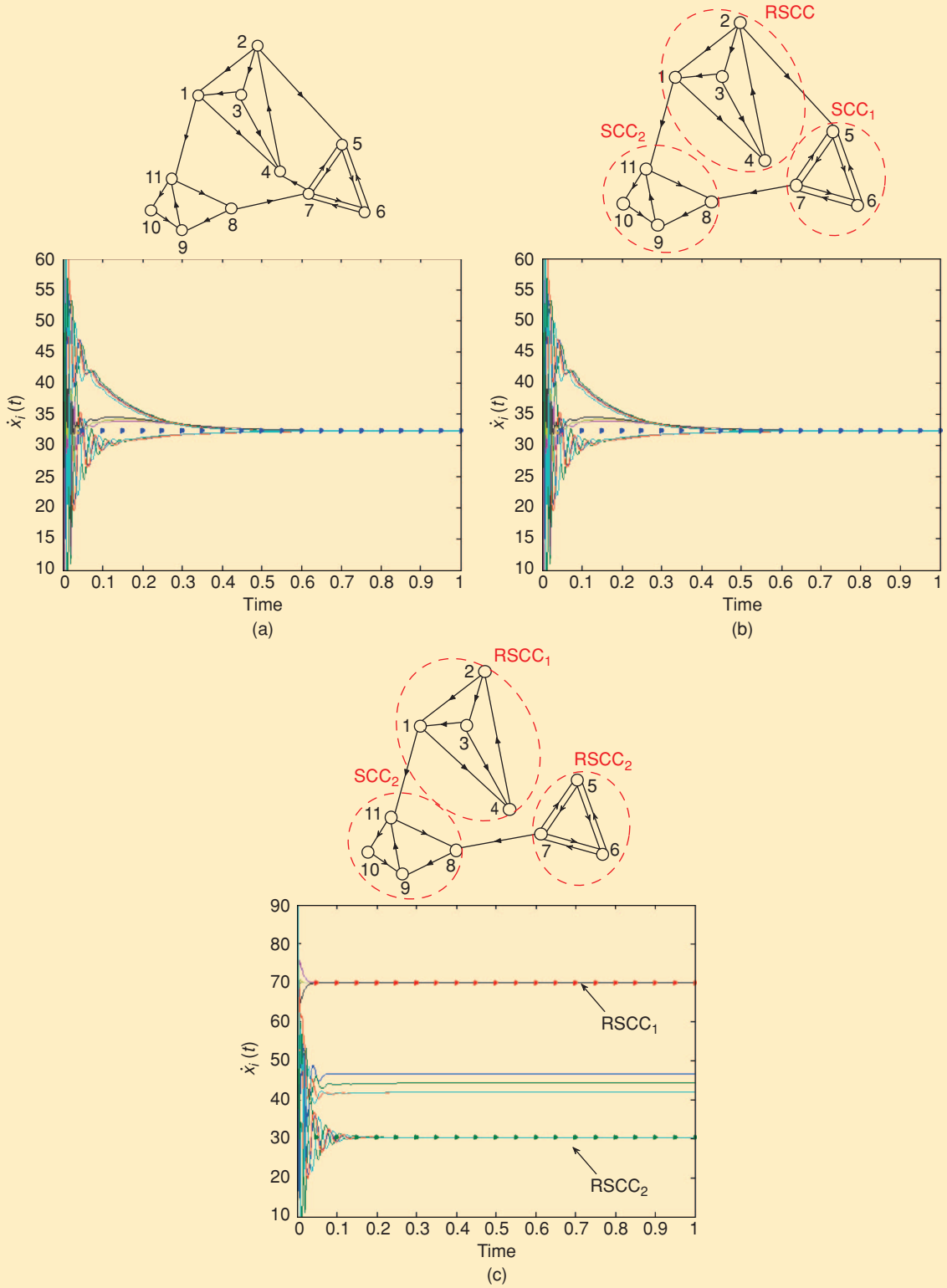
If we denote by 1 and 0 the vectors of all ones or zeros, respectively, it is straightforward to verify that $\mathbf{L}1 = 0$; i.e., zero is an eigenvalue of $\mathbf{L}$ corresponding to the right eigenvector 1. The multiplicity of the zero eigenvalue of $\mathbf{L}$ is equal to the minimum number of directed trees contained in a directed spanning forest of $\mathcal{G}$. Moreover, the zero eigenvalue of $\mathbf{L}$ is simple if and only if $\mathcal{G}$ contains a spanning directed tree or, equivalently, $\mathcal{G}$ is QSC.

FORMS OF CONSENSUS

We can now illustrate the achievable forms of consensus. We will consider the linear coupling case first, with or without propagation delays, and then we will consider the nonlinear coupling case, with no delays.

Linear Coupling

In the linear coupling case, i.e., when $h(x) = x$, system (4) has been proved to synchronize *if and only if* the digraph is QSC and the consensus value is [28]



[FIG1] Examples of graphs: (a) Strongly connected graph. (b) Quasi strongly connected graph with one root strongly connected component and two strongly connected components. (c) WC graph containing a two-tree forest (top row) and evolution of the state derivatives of (4) (bottom row).

$$\lim_{t\to\infty} \dot{x}_q(t) = \omega^* = \frac{\sum_{i=1}^{N} \gamma_i c_i g_i(y_i)}{\sum_{i=1}^{N} \gamma_i c_i + K \sum_{i=1}^{N} \sum_{j=1}^{N} \gamma_i a_{ij} \tau_{ij}}, \quad \forall q, \tag{6}$$

where γ_i are the entries of the left eingenvector associated to the null eigenvalue of the graph Laplacian. The coefficient γ_i is positive if and only if node i can reach every other node through a strong path, otherwise $\gamma_i = 0$. This means that the only nodes that have an impact on the final consensus are those that can reach *every other node* in the network, even if this occurs through several hops. As an extreme case, the final consensus coincides with the decision taken by a *single* node, let us say node r, if the graph contains only one directed tree whose root node is node i. More generally, if the graph is not even QSC but it contains a forest with K strongly connected root components, the system evolution gives rise to K clusters, where the nodes belonging to each cluster achieve a local (within the cluster) consensus [28]

$$\lim_{t\to\infty} \dot{x}_q(t) = \omega_q^* = \frac{\sum_{i\in\mathcal{N}_i} \gamma_i c_i g_i(y_i)}{\sum_{i\in\mathcal{N}_i} \gamma_i c_i + K \sum_{i\in\mathcal{N}_i} \sum_{j=1}^{N} \gamma_i a_{ij} \tau_{ij}}, \quad \forall q \in \mathcal{C}_k, \quad k = 1, \ldots, K, \tag{7}$$

where $\mathcal{C}_k$ indicates the kth cluster. Hence, the same mechanism leading to a global consensus, as in (6), is also able to produce clusters, as in (7), depending on the network connectivity properties. In practice, we may force one behavior or the other, acting on the transmit powers p_j; i.e., on the coefficients a_{ij}, as they ultimately determine the graph topology.

The consensus forms described before are appealing because they can be achieved under very broad conditions. However, they present also a drawback due to their dependence on the network topology, through the weights γ_i, and the channel status, through the coefficients a_{ij} and the delays τ_{ij}. It was proved in [28] that the dependence of the final consensus on the topology disappears (in the sense that $\gamma_i = 1$) if and only if the graph is balanced; i.e., $\sum_{j=1}^{N} a_{ij} = \sum_{j=1}^{N} a_{ji}, \forall i$.

Note that for undirected strongly connected graphs, this condition is always true. Then, a topology-independent consensus is achievable by changing the channel coefficients in order to get a balanced graph. Recalling, from (4), that $a_{ij} = \sqrt{p_j |h_{ij}|^2 / d_{ij}^2}$, in time division duplexing systems, where $h_{ij} = h_{ji}$, a balanced graph is obtained by using the same transmit power on each node, so that $p_j = p, \forall j$, and thus $a_{ij} = a_{ji}, \forall\, i$ and j.

Even if we are able to get rid of the topology dependence, the final consensus values, as given in (6) or (7), still depend on the channel parameters. Actually, this is true only in the delayed case, because for $\tau_{ij} = 0$ the channel dependence disappears. But in the delayed case, this dependence prevents expressions (6) or (7) from becoming optimal decision statistics, as these should depend only on the gathered data. Nevertheless, also in the delayed case, we can easily eliminate the channel-dependent term, in the denominator of both (6) or (7), using the following two-step procedure. We run the evolution equation first with the right functions $g_i(y_i)$, thus obtaining the consensus value $\omega^*(\mathbf{y})$. Then, we run the algorithm again, this time setting $g_i(y_i) = 1$, thus obtaining a second consensus, let us say $\omega^*(\mathbf{1})$. It is easy to verify, from (6) or (7), that the ratio $\omega^*(\mathbf{y})/\omega^*(\mathbf{1})$ assumes a value independent of the channel coefficients. If we also insure that the graph is balanced, the previous procedure allows each node to reach, asymptotically through its state derivative, any form of consensus that can be expressed in the form [28]

$$f(y_1, y_2, \ldots, y_N) = u\left(\left(\sum_{i=1}^{N} c_i\right)^{-1} \sum_{i=1}^{N} c_i g_i(y_i)\right), \tag{8}$$

where $u(\cdot)$ is a function that each node can apply to the ratio achieved through (6) or (7).

Similarly, in the vector case (5), the final consensus, in the absence of delays, assumes the form [28]

$$\mathbf{f}(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N) = \mathbf{u}\left[\left(\sum_{i=1}^{N} \mathbf{C}_i\right)^{-1} \sum_{i=1}^{N} \mathbf{C}_i \mathbf{g}_i(\mathbf{y}_i)\right]. \tag{9}$$

The forms of consensus, (8) or (9), although apparently very specific, incorporate several cases of practical interest in statistical signal processing. For example, the vector form (9) may be used to solve Problem 1 as formulated in the Motivating Examples section above; i.e., to achieve the globally optimal ML estimate, given by (2), simply by running (5), after setting $\mathbf{C}_i = \mathbf{A}_i^T \mathbf{R}_i^{-1} \mathbf{A}_i$ and $\mathbf{g}_i(\mathbf{y}_i) \triangleq (\mathbf{A}_i^T \mathbf{R}_i^{-1} \mathbf{A}_i)^{-1} \mathbf{A}_i^T \mathbf{R}_i^{-1} \mathbf{y}_i$.

Similarly, we may also solve Problem 2, using (8). The MAP test may in fact be constructed by evolving (4) with $c_i = 1$ and $g_i(\mathbf{y}_i/\mathcal{H}_k) = \log(p_i(\mathbf{y}_i/\mathcal{H}_k)P(\mathcal{H}_k))$ for K times (i.e., as many time as the number of hypotheses). At convergence, each node applies the function $u(\omega_k^*) = \exp(\omega_k^*)$ to the consensus achieved under hypothesis k. It is easy to verify that each final value $u(\omega_k^*)$ is proportional to the argument of the MAP detector in (3).

Nonlinear Coupling Case

The nonlinear coupling case is of course more difficult to analyze mathematically. Nevertheless, in [27], it was shown that, in the case of *undirected* graphs, if the coupling function $h(\cdot)$ in (4) or (5) is an odd, monotonically increasing function, then the network achieves a global consensus, still in the form of (8) or (9), if the global coupling coefficient K is greater than a critical value. This value was upper-bounded by the following expression [27]:

$$K_U = \frac{2\|\mathbf{D}_c \Delta\omega\|_2}{h_{\max}\, \lambda_2(\mathbf{L}_A)},$$

where $\mathbf{D}_c = \mathrm{diag}(c_1, \ldots, c_N)$, $\Delta\omega = [g_1(y_1), g_2(y_2), \ldots, g_N(y_N)]^T - \omega^* \mathbf{1}$, $h_{\max}$ is the maximum of $h(x)$; $\lambda_2(\mathbf{L}_A)$ is the

so called *algebraic connectivity* of the graph; i.e., the second smallest eigenvalue of the graph Laplacian $\mathbf{L}_A$. The same condition applies to the vector observation case. Hence, at least in the absence of delays and for symmetric links, a global consensus toward the optimal decision statistics is still achievable, provided that the nonlinear coupling function is an antisymmetric increasing function and that the coupling is sufficiently strong. Interestingly, the nonlinear case has a greater variety of behaviors than the linear case as the network may be designed to avoid the convergence toward a unique common value simply choosing K smaller than a critical value [27]. This property might be used, for example, to achieve some forms of clustering.

NUMERICAL EXAMPLES

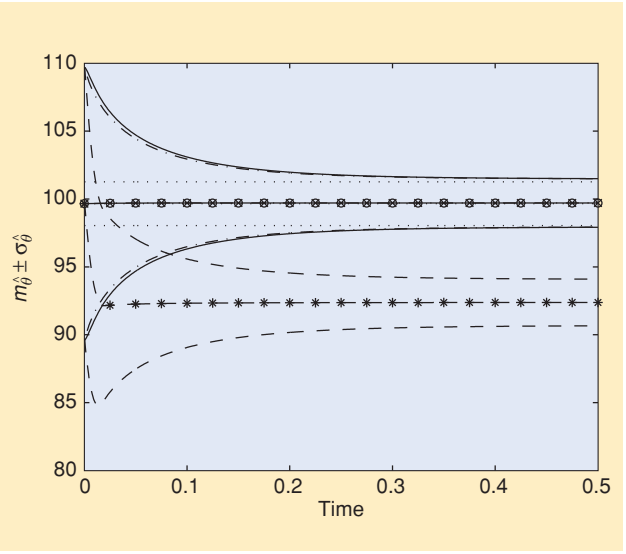
We now present some numerical examples to validate the theoretical findings described in the previous sections.

EXAMPLE 1

In the bottom row of Figure 1, we plot the dynamical evolutions of the state derivatives of (7) versus time for the three network topologies sketched in the top row. Each curve shows also the theoretical asymptotic values of (6) or (7) (dashed line with arrows). As predicted by the theory, the dynamical system of Figure 1(a) achieves a global consensus, since the underlying digraph is SC. The network of Figure 1(b), instead, is not SC, but it is QSC and the system is still able to globally synchronize, since there is a set of nodes in the RSCC component able to reach all the other nodes. However, the final consensus in such a case contains only the contributions from the nodes in the RSCC, since no other node belongs to the root of a spanning-directed tree of the condensation digraph. Finally, the system of Figure 1(c) cannot reach a global consensus since there is no node that can reach all the others, but it does admit two disjoint clusters, corresponding to the two RSCCs; namely, RSCC1 and RSCC2. The middle lines of Figure 1(c) refer to the nodes of the SCC component that do not belong to either RSCC1 or RSCC2. These nodes are affected by the consensus achieved by the two RSCC components, but they cannot affect them. Observe that, in all the cases, the state derivatives of the (global or local) clusters converge to the values predicted by the closed-form expression given in (6) or (7), depending on the network topology and the channel parameters.

EXAMPLE 2

The behaviors shown in the previous example refer to a given realization of the topology, with given link coefficients and observations. In this example, we report the performance obtained in the estimation of a scalar variable, averaged over 100 independent realizations of network topology, channel coefficients, and noise terms. Each sensor observes a variable $y_i = A_i \xi + v_i$, where v_i is additive zero mean Gaussian noise, with variance σ_i^2. The goal is to estimate ξ. The estimate is performed through the interaction system (4), setting $g_i(y_i) = y_i/A_i$ and $c_i = A_i^2/\sigma_i^2$, in order to achieve the globally optimal ML estimate. The network is composed of 40 nodes, randomly spaced over a square of size D. The analog system (4) is implemented in discrete time, with sampling time $T_s = 10^{-3}$ s. The size of the square occupied by the network is chosen in order to have a maximum delay $\tau_{\max} = 100\ T_s$. To simulate a practical scenario, the channel coefficients a_{ij} are generated as independent and identically distributed Rayleigh random variables, to accommodate for channel fading. Each variable a_{ij} has a variance depending on the distance d_{ij} between nodes i and j, equal to $\sigma_{ij}^2 = p_j/(1 + d_{ij}^2)$. We set the threshold on the amplitude of the minimum useful signal to zero, so that, at least in principle, each node hears each other node. The corresponding graph is then SC. Figure 2 shows the estimated average of the state derivative (plus and minus the estimation standard deviation), as a function of the iteration time. The results refer to following cases of interest: 1) ML estimate achieved with a centralized system, with no communication errors between nodes and fusion center (parallel dotted lines); 2) estimate achieved with the proposed method, with no propagation delays, as a benchmark term (dashed and dotted lines plus x marks for the average value); 3) estimate achieved with the proposed method, with propagation delays (dashed lines plus * for the average value); 4) estimate achieved with the two-step estimation method leading to the ratio $\omega^*(\mathbf{y})/\omega^*(\mathbf{1})$ (solid lines plus circles for the average value). From Figure 2, we can see that, in the absence of delays, the (decentralized) iterative algorithm based on (4) behaves, asymptotically, as the (centralized) globally optimal ML estimator. In the presence of delays, we observe a clear bias (dashed lines), due to the large delay values but still with a final estimation variance close to the ML estimators. Interestingly, the two-step procedure provides results very close to the optimal ML estimator, with no apparent bias, in spite of the large delays and the random channel fading coefficients.



[FIG2] Estimated value standard deviation.

NETWORK TOPOLOGY AND ENERGY EXPENDITURE

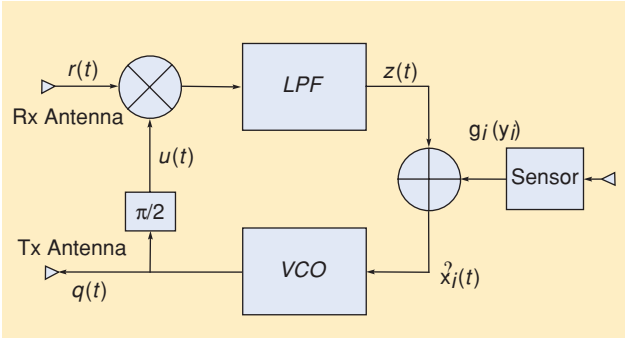
The totally distributed approach used to achieve a distributed consensus is appealing because it does not need a fusion center. However, there is penalty to be paid due to the iterative process necessary to achieve the consensus. The overall energy necessary to achieve the final decision is the sum of the powers transmitted by each sensor multiplied by the convergence time. As shown in [26], the convergence rate of the proposed algorithm is proportional to $K\lambda_2(\mathbf{L}_A)$, where $\lambda_2(\mathbf{L}_A)$ is the *algebraic connectivity*. In the simple case in which all nodes transmit with the same power p_T, the total energy consumption is [32]

$$\mathcal{E}(p_T) = \alpha \frac{Np_T}{\lambda_2(\mathbf{L}_A(p_T))}, \tag{10}$$

where with $\lambda_2(\mathbf{L}_A(p_T))$ we make explicit the dependence of the algebraic connectivity on the single node transmit power p_T, through the Laplacian; α is a multiplicative factor determined by the requirement on the final accuracy. Interestingly, we may observe that the higher the transmit power, the higher is $\lambda_2(\mathbf{L}_A)$ and hence the faster is the convergence. However, at the same time, the higher is also the numerator of (10). Hence, there exists, in general, an optimal p_T resulting in the best trade-off between the local transmit power and the convergence time, as shown in [32].

IMPLEMENTATION ISSUES

An example of implementation based on a phase-locked loop (PLL) is illustrated in Figure 3. The received signal is $r(t) = \sum_{j=1}^{n} a_{ij}\cos(2\pi f_0 t + x_j(t))$, where f_0 is the nominal carrier frequency. The received signal is beaten with the reference signal $u(t) = \sin(2\pi f_0 t + x_i(t))$ obtained as a 90-degree phase-shifted version of the output of the voltage-controlled oscillator (VCO), $q(t) = \cos(2\pi f_0 t + x_i(t))$, which is also the transmitted signal. The low-pass filter removes the components with spectrum centered around $2f_0$ and its output is $z(t) = \frac{K}{c_i}\sum_{j=1}^{n} a_{ij}\sin(x_j(t) - x_i(t))$. It is easy to check that the input of the VCO is a special case of (4), corresponding to $h(x) = \sin(x)$. Hence, a *single PLL* scheme, plus the associated sensing device, is sufficient for implementing the *whole* node evolving according to the interaction model (4). What is also important to observe is that it is not necessary for the receiver to discriminate the signals transmitted by different sensors, as opposed to the conventional average consensus techniques: What is really necessary is that the receiver be able to get the summation given by $z(t)$. As a consequence, the medium access control may be implemented in a very simple way.



[FIG3] Implementation of (4) as a phase-locked loop.

An alternative implementation may be based on impulse radio technology. In such a case, every node transmits a periodic sequence of pulses, with rate $1/T$. If we take the discrete version of (4), sampled at a rate $1/T_s$, with $T_s \ll T$, that is,

$$x_i[n+1] = x_i[n] + T_s g_i(y_i) + \frac{KT_s}{c_i}\sum_{j=1}^{N} a_{ij}\, h(x_j[n - n_{ij}] - x_i[n]) + w_i[n], \quad i = 1, 2, \ldots, N,$$

we can associate the *pulse position* $t_i[n]$ of the ith oscillator, within the nth period, to the state variable $x_i[n]$. In this case, consensus, as used in this article, refers to the situation in which all nodes reach the same interpulse time $t_i[n+1] - t_i[n]$.

KEY CHALLENGES FOR FUTURE RESEARCH

In conclusion, the use of self-synchronization mechanisms appears to possess great potential for devising simple and effective schemes for implementing globally optimal decentralized decision systems. One nice feature of the proposed approach is the possibility to make the network converge either to a global, common set of parameters or to form clusters of consensus by properly devising the network connectivity. The approach is useful also for self-synchronization of wide area networks, where the propagation delays cannot be neglected. The self-synchronization capability might be especially useful if merged with type-based multiple access schemes to get coherent transmissions. A key parameter affecting the performance of decentralized algorithms is the algebraic connectivity, as this influences both the energy spent by the network to achieve consensus and the convergence time. The latter parameter determines the maximum rate of change of the monitored phenomenon that the network may be able to track. An especially important contribution to this area would be the incorporation of the properties of random graphs, possibly with a clear physical motivation, into the analysis of the convergence capabilities. Moreover, more complicated tasks than just estimation of a set of parameters or multiple hypothesis testing should be addressed with, probably, more sophisticated interaction mechanisms. Finally, we note that further cross fertilization between mathematical biology and distributed signal processing, an example of which has been presented in this article, has great potential for future emerging applications in distributed systems.

ACKNOWLEDGMENTS

This work has been partially funded by the WINSOC Project, a Specific Targeted Research Project (contract number 0033914) co-funded by the INFSO DG of the European Commission within the RTD activities of the Thematic Priority Information Society Technologies, and by ARL/ERO, contract number N62558-05-P-0458.

AUTHORS

Sergio Barbarossa (sergio@infocom.uniroma1.it) received the M.Sc. in 1984 and the Ph.D. in electrical engineering in 1988, both from the University of Rome "La Sapienza," Rome, Italy. He joined the Radar System Division of Selenia in 1985 as a Radar System Designer. In 1987, he was a research engineer at the Environmental Research Institute of Michigan (ERIM), Ann Arbor, Michigan. From 1988 until 1991, he was an assistant professor at the University of Perugia. In November 1991, he joined the University of Rome "La Sapienza," where he is now a full professor and director of graduate studies. He has held positions as visiting scientist and visiting professor at the University of Virginia (1995 and 1997), the University of Minnesota (1999), and the Polytechnic University of Catalonia (2001). He has participated in international projects on synthetic aperture radar systems, flown on the space shuttle, and he has been the primary investigator in four European projects on multiantenna systems and multihop systems. He is currently the scientific responsible of the European project WINSOC (on wireless sensor networks), and he is one of the primary investigators of the European project SURFACE (on reconfigurable air interfaces for wideband multiantenna communication systems). From 1998 to 2004, he was a member of the IEEE Signal Processing for Communications Technical Committee. He served as an associate editor of the *IEEE Transactions on Signal Processing* from 1998 to 2001 and from 2004 to 2006. He has co-edited a special issue of the *EURASIP Journal of Applied Signal Processing* on MIMO Communications and Signal Processing and a special issue of the *IEEE Journal on Selected Areas in Communications* on Optimization of MIMO Transceivers for Realistic Communication Networks: Challenges and Opportunities. He received the 2000 IEEE Best Paper Award from the IEEE Signal Processing Society for a co-authored paper in the area of signal processing for communications. He is the author of a research monograph titled "Multiantenna Wireless Communication Systems." His current research interests lie in the areas of sensor networks, cooperative communications, and distributed decision.

Gesualdo Scutari (scutari@infocom.uniroma1.it) received the electrical engineering and Ph.D. degrees from the University of Rome "La Sapienza," Rome, Italy, in 2001 and 2004, respectively. During 2003, he held a visiting research appointment at the Department of Electrical Engineering and Computer Sciences at the University of California at Berkeley. He currently holds a post-doc position at the INFOCOM Department at the University of Rome "La Sapienza." He has participated in two European projects on multiantenna systems and multihop systems (IST SATURN and IST-ROMANTIK) and he is currently involved in the European projects WINSOC, on wireless sensor networks, and SURFACE, on reconfigurable air interfaces for wideband multiantenna communication systems. His primary research interests include sensor networks and communication aspects of wireless multiple-input multiple-output (MIMO) channels, with special emphasis on convex optimization theory and game theory applied to communications systems.

REFERENCES

[1] Z.-Q. Luo, M. Gastpar, J.J. Liu, and A. Swami, Eds., "Special issue on distributed signal processing in sensor networks," *IEEE Signal Processing Mag.*, vol. 23, no. 4, July 2006.

[2] E.J. Duarte-Melo and M. Liu, "Data-gathering wireless sensor networks: Organization and capacity," *Comput. Netw.*, vol. 43, pp. 519–537, Apr. 2003.

[3] A. Giridhar and P.R. Kumar, "Computing and communicating functions over sensor networks," *IEEE J. Selected Areas Commun.*, vol. 23, pp. 755–764, Apr. 2005.

[4] A. Giridhar and P.R. Kumar, "Toward a theory of in-network computation in wireless sensor networks," *IEEE Commun. Mag.*, vol. 44, pp. 98–107, Apr. 2006.

[5] D.P. Bertsekas and J.N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, 2nd ed. Nashua, NH: Athena Scientific, 1989.

[6] N.A. Lynch, *Distributed Algorithms*. San Mateo, CA: Morgan-Kaufmann, 1997.

[7] G. Mergen and L. Tong, "Type based estimation over multiaccess channels," *IEEE Trans. Signal Processing*, vol. 54, pp. 613–626, Feb. 2006.

[8] E. Eisenberg and D. Gale, "Consensus of subjective probabilities: The pari-mutuel method," *Ann. Mathemat. Stat.*, vol. 30, pp. 165–168, Mar. 1959.

[9] R. Olfati-Saber and R.M. Murray, "Consensus protocols for networks of dynamic agents," in *Proc. 2003 American Control Conf.*, June 2003, pp. 951–956.

[10] 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.

[11] C.W. Wu, "Agreement and consensus problems in groups of autonomous agents with linear dynamic," in *Proc. ISCAS 2005*, May 2005, pp. 292–295.

[12] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. IPSN 2005*, Apr. 2005, pp. 63–70.

[13] R. Olfati-Saber, J.A. Fax, and R.M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, to be published.

[14] V. Saligrama, M. Alanyali, and O. Savas, "Distributed detection in sensor networks with packet losses and finite capacity links," *IEEE Trans. Signal Processing*, vol. 54, pp. 4118–4132, Nov. 2006.

[15] R. Olfati-Saber, E. Franco, E. Frazzoli, and J.S. Shamma, "Belief consensus and distributed hypothesis testing in sensor networks," in *Workshop Network Embedded Sensing and Control*, South Bend, IN, Oct. 2005, pp. 169-182.

[16] M. Alanyali, V. Saligrama, O. Savas, and S. Aeron, "Distributed Bayesian hypothesis testing in sensor networks," in *Proc. American Control Conf.*, Boston, MA, June 2004, pp. 5369–5374.

[17] A. Jadbabaie, J. Lin, and A.S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbour rules," *IEEE Trans. Automat. Contr.*, vol. 48, pp. 988–1001, June 2003.

[18] S.M. Kay, *Fundamentals of Statistical Signal Processing*. Upper Saddle River, NJ: Prentice Hall, 1998.

[19] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization—A Universal Concept in Nonlinear Sciences*. Cambridge, UK: Cambridge Univ. Press, 2001.

[20] R. Mirollo and S.H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, pp. 1645–1662, 1990.

[21] F.C. Hoppensteadt and E.M. Izhikevich, "Thalamo-cortical interactions modeled by weakly connected oscillators: Could brain use FM radio principles?" *BioSystems*, vol. 48, pp. 85–94, 1998.

[22] E.M. Izhikevich, "Weakly pulse-coupled oscillators, FM interactions, synchronization, and oscillatory associative memory," *IEEE Trans. Neural Networks*, pp. 508–526, May 1999.

[23] Y.-W. Hong, L.F. Cheow, and A. Scaglione, "A simple method to reach detection consensus in massively distributed sensor networks," in *Proc. ISIT 2004*, June 2004, p. 250.

[24] D. Lucarelli and I.J. Wang, "Decentralized synchronization protocols with nearest neighbor communication," in *SenSys 2004*, Baltimore, MD, Nov. 2004, pp. 62-68.

[25] S. Barbarossa, "Self-organizing sensor networks with information propagation based on mutual coupling of dynamic systems," *IWWAN 2005*, London, UK, May 2005.

[26] S. Barbarossa and G. Scutari, "Decentralized maximum likelihood estimation for sensor networks composed of nonlinearly coupled dynamical systems," *IEEE Trans. Signal Processing*, to be published.

[27] S. Barbarossa, G. Scutari, and L. Pescosolido, "Global stability of a population of mutually coupled oscillators reaching global maximum likelihood estimate through a decentralized approach," *ICASSP 2006*, Toulouse, France, May 2006, pp. IV-385-388.

[28] G. Scutari, S. Barbarossa, and L. Pescosolido, "Distributed decision through self-synchronizing sensor networks in the presence of propagation delays and non-reciprocal channels," *IEEE Trans. Signal Processing*, submitted for publication.

[29] G. Scutari, S. Barbarossa, and L. Pescosolido, "Optimal decentralized estimation through self-synchronizing networks in the presence of propagation delays," *SPAWC 2006*, Cannes, France, July 2006.

[30] Y. Kuramoto, *Chemical Oscillations, Waves, and Turbulence*. Berlin: Springer-Verlag, 1984.

[31] R.A. Brualdi and H.J. Ryser, *Combinatorial Matrix Theory*. Cambridge, UK: Cambridge Univ. Press, 1991.

[32] S. Barbarossa, G. Scutari, and A. Swami, "Achieving consensus in self-organizing wireless sensor networks: The impact of network topology on energy consumption," in *Proc. ICASSP 2007*, Honolulu, HI, Apr. 2007, to be published. SP