On In-Network Computation via Wireless Multiple-Access Channels with Applications

(Invited Paper)

Sławomir Stańczak, Mario Goldenbaum, Renato L. G. Cavalcante and Federico Penna

Fachgebiet Informationstheorie und theoretische Informationstechnik,
Technische Universität Berlin, Einsteinufer 25, 10587 Berlin, Germany

Abstract—**Wireless sensor networks have increasingly been gaining importance because of their applications in many domains, including remote sensing, traffic management, disaster alarm, and anomaly detection. In many of these applications, sensor nodes cooperate in obtaining an estimate of a control or environmental parameter to achieve capabilities that cannot be achieved by letting the nodes work individually. Cooperation may take on many forms, but it usually involves the computation of weighted averages, maximum values, or polynomial functions of sensor readings. As the readings are available at spatially distributed nodes, wireless communication resources have to be used as efficiently as possible in the computation of the underlying functions. In this paper, we present novel techniques that exploit recent ideas on analog computation coding over wireless channels and show how the proposed techniques can be applied in real-world estimation problems.**

I. INTRODUCTION

Wireless networks are increasingly being required to cope with nodes suffering from strong interference and having very limited computational and energy resources. This fact poses serious challenges for system designers, who have to conciliate conflicting requirements: high network availability, long network lifetime, and high data rate. The difficulty in addressing these challenges is further exacerbated by the fact that in many envisioned systems, such as those involving sensor networks, nodes have to harvest energy from unreliable sources such as sunlight or vibrations. Solutions for these challenges typically require the collection of network parameters to define optimal strategies with subsequent transmissions.

In many systems, after collecting these network system state information, users/nodes typically have to compute functions of network parameters (e.g., signal-to-interference+noise ratios, measurement values, utilities) to define rules for increasing the system performance [1]. To date, system engineers typically work with the philosophy of strictly separating the processes of *communication* and *computation*, and, in such a paradigm, interference caused by multiple users occupying the same channel is considered a problem to be mitigated. However, in stark contrast with the above paradigm, the work in [2] shows that the superposition property of wireless channels can be used to merge the processes of communication and computation, which leads to drastic improvements in

system throughput and energy consumption. Such an approach is particularly suitable for estimation tasks involving sensor networks. For example, [3] and [4] describe analog joint source-channel communication schemes that exploit a match between the interference property of the Gaussian MAC and the optimal estimator for extracting a desired parameter from a collection of noisy sensor readings. The approach outperforms their digital counterparts based on the standard separation design principle (originally proposed in the landmark paper [5]) between source and channel coding. Extensions of the analog source-channel communication scheme to more general estimation tasks are found in [6]–[8], and adaptations to detection problems have been proposed in [9]–[11].

In addition to estimation problems, interference among multiple users in wireless channels is also intelligently exploited in a technique known as physical-layer network coding. In contrast to traditional network coding methods (which are designed on the network layer), physical-layer network coding generates linear codewords immediately on the physical layer by making nodes transmit simultaneously, thus superposing the transmitted electromagnetic waves at the receivers [12]–[14]. This principle can be also applied to bi-directional relaying systems where users exchange data by means of a relay node located between the users exchanging information. In such schemes, the relay node has the possibility of applying different strategies to the incoming superimposed users' data. It can, for example, simply amplify and forward the receive signal to the users (amplify-and-forward), or it can decode and re-encode individual messages before forwarding them (decode-and-forward). More recently, in an alternative novel strategy titled "compute-and-forward" [15], users apply a common code, and relays decode *linear equations* of the transmitted code words only. After decoding, the relays forward those equations to the receiving ends, which are able to decode the transmitted messages if enough equations are received.

Common to the above schemes for estimation and network coding is the fact that they can be viewed as particular instances of distributed function computation problems. Hence, there is a clear need for corresponding efficient algorithms because they are a fundamental building block of distributed in-network computation. In particular, an important class of strongly related algorithms are consensus algorithms, which in recent years have received a great deal of attention by the control and signal processing communities [16]–[18]. The

main objective of these algorithms is to compute weighted averages of values reported by nodes in a network. To date, most consensus algorithms do not exploit all physical properties of wireless channels in their full extent; they typically broadcast and fuse in hardware their reported values in a decentralized fashion. Therefore, there are opportunities to improve consensus algorithms by using ideas similar to those used in physical-layer network coding, where the processes of computation and communication are merged.

This paper summarizes our results on analog computation that merges the processes of computation and communication. In Section III, we point out several potential applications to real-world problems where the analog computation approach can be used for distributed in-network computation.

II. ANALOG COMPUTATION VIA SIGNAL POWERS

To explain the basic idea of analog computation via signal powers, let us assume K spatially distributed transmitter-receiver pairs. Further assume that each receiver, say receiver k, aims at computing the function $f_k(X_1, \ldots, X_K)$ where $X_l \geq 0$ is a variable/measurement[1] available only at transmitter l. Now, the idea of the analog computation scheme is to let all transmitters concurrently send random sequences of zero-mean independent symbols with the transmission power of the kth transmitter being proportional to X_k. During the transmission, each receiver, say receiver k, estimates the function value from the received power by averaging over all symbol intervals. Good estimates are possible if the impact of the fading channel and receiver noise on the received signal is eliminated and the random sequences are sufficiently long so that the received power is well-approximated by the sum of the transmission powers. As explained later, this sum essentially contains all the information needed to compute *any* function.

The advantages of the scheme are the following:

1) No perfect synchronization at the symbol and phase level between different transmitters is required. In fact, a coarse frame synchronization is sufficient in the sense that the overlap of different sequences in time is significant when compared to the duration of the sequences.
2) Except for the coarse synchronization, there is no need for cooperation between different nodes; each random sequence can be generated locally without any coordination.

The main challenge is to overcome the fading effects of the wireless channel. In some cases, this effect can be removed by an appropriate channel inversion prior to transmission, in which case the transmission powers also depend on channel realizations. This will be illustrated in the next section, where we consider a distributed computation of search directions for power control in distributed wireless networks with K interfering transmitter-receiver pairs.

[1]The nonnegativity assumption does not impact the applicability of the approach since in practical settings it is always possible to find a bijective map from the measurement space on a connected set of nonnegative reals.

A. Analog Computation for Power Control

Power control is a central mechanism for resource allocation and interference management in wireless networks (see [19] and references therein). This is especially true for the interference channel with K transmitter-receiver pairs that form K communication links. Each transmitter transmits its independent data to the corresponding receiver at power p_k. Each receiver employs a single-user decoder and treats the interference from other links/pairs as noise. Now, the objective of power control is to find a power allocation $\mathbf{p} = (p_1, \ldots, p_K) \in \mathbb{R}_+^K$ so as to maximize a given aggregate utility function:

$$U(\mathbf{p}) := \sum\nolimits_{k=1}^{K} w_k \theta(\text{SIR}_k(\mathbf{p})), \quad \forall_k \, w_k > 0 \qquad (1)$$

where SIR_k is the signal-to-interference+noise ratio of link k, w_k is a given weight and $\theta : \mathbb{R}_+ \to \mathbb{R}$ is a twice continuously differentiable, strictly increasing function.

If the inverse function of θ is log-convex, then the problem can be formulated as a convex optimization problem and solved efficiently by an iterative method of the following form: $\mathbf{p}(n + 1) = \Pi[\mathbf{p}(n) + s(n)\Delta U(\mathbf{p}(n))]$, where Π denotes a projection operator to satisfy some constraints, $s(n)$ is a suitable step size sequence and $\Delta U : \mathbb{R}_+^K \to \mathbb{R}^K$ is used to denote a search direction that depends on the algorithmic solution at hand. Widely-used algorithms include Newton methods, the steepest decent method, the BFGS method and gradient methods. In all these cases, an efficient *distributed* computation of the search direction poses a significant challenge because in general each component of ΔU depends on all local variables distributed over the entire network. The difficulty results from the interference coupling, which is described by the so-called gain matrix. To define the matrix, let $V_{k,l} \geq 0, l \neq k$, be the power gain between transmitter l and the output of receiver k. In other words, if the transmit power of link l is p_l, then the expected interference power caused by this link at the output of receiver $k \neq l$ is $p_l V_{k,l}$. $V_{k,l}$ depends on the spectrum, receiver structure, the channel state and other system parameters. Consequently, $V_{k,k} > 0$ is the power gain on the desired link k. The nonnegative matrix $(\mathbf{V})_{k,l} = v_{k,l} = V_{k,l}/V_{k,k}, k \neq l$ with $\text{trace}(\mathbf{V}) = 0$ is referred to as the gain matrix. We assume that all links are reciprocal with respect to powers which is significantly weaker than the usual channel reciprocity, in which case the phases of the complex channel coefficients must coincide as well.

It turns out that the gain matrix plays a crucial role when computing the search direction. To see this, we focus on the standard gradient-projection method, in which case we have $\Delta U(\mathbf{p}) = -\nabla U(\mathbf{p})$ where the gradient vector can be shown to be given by [19]

$$\nabla U(\mathbf{p}) = \mathbf{g}(\mathbf{p}) - \mathbf{V}^T \mathbf{q}(\mathbf{p}) \qquad (2)$$

where $\mathbf{g}(\mathbf{p}) := (g_1(\mathbf{p}), \ldots, g_K(\mathbf{p}))$ and $\mathbf{q}(\mathbf{p}) := (q_1(\mathbf{p}), \ldots, q_K(\mathbf{p}))$ are vectors of some local variables so that $g_k(\mathbf{p})$ and $q_k(\mathbf{p})$ are available only on link k. Thus, the main problem lies in computing $\mathbf{V}^T \mathbf{q}(\mathbf{p})$ for some given $\mathbf{p}$ as $\mathbf{V}$ is

neither diagonal nor symmetric. Furthermore, the gain matrix is not known at any node since its entries depend on the channel coefficients of the interference links, which are usually not estimated in wireless networks. Therefore, we proposed in [20]–[22] to transmit sequences of power-encoded random symbols over an adjoint network defined as follows.

Definition 1. Consider a network with K links and the gain matrix $\mathbf{V}$. Any network with K links and the gain matrix $\mathbf{U} \in \mathbb{R}_+^{K \times K}$ is adjoint to the network if $\mathbf{U} = \mathbf{V}^T$.

In what follows, let a network with the gain matrix $\mathbf{V}$ be given. In order to see how to obtain a corresponding adjoint network, we introduce the notion of reversed network.

Definition 2. We call a network reversed if the roles of transmitters and receivers on each link in a (primal) network are reversed.

By the reversed (or swapped) roles we mean that, in each transmitter–receiver pair, say the pair on link k, the kth transmitter becomes the kth receiver and vice versa. The corresponding link in a reversed network is labeled by k.

It can be easily verified that the gain matrix of the reversed network is not equal to the transposed gain matrix $\mathbf{V}^T$. However, a straightforward examination shows [22] that $\mathbf{V}^T$ is obtained if each transmitter in a reversed network inverts its "own" channel such that the resulting signal power gain between each transmitter–receiver pair is equal to 1 (i.e., $V_{k,k} = 1$). We summarize these observations in a theorem.

Theorem 1 ([22]). A reversed network is adjoint to a given network if each transmitter in the reversed network, say transmitter k, $k = 1, \ldots, K$, inverts its channel by multiplying transmit symbols by $1/\sqrt{V_{k,k}} > 0$.

When each transmitter in an adjoint network, say transmitter k, transmits a sequence of random symbols (over the adjoint network) at power q_k (known only to the kth receiver), then, in the absence of noise and sufficiently long random sequences, an average signal power observed at the lth receiver is approximately equal to $((\mathbf{I} + \mathbf{V}^T)\mathbf{q})_l$, which is sufficient to compute the gradient vector (2). Note that the transmission over an adjoint network involves 1) swapping the roles of transmitter and receivers, and 2) channel inversions as specified in Theorem 1. In the presence of noise $\mathbf{n}$, the average signal power becomes $((\mathbf{I} + \mathbf{V}^T)\mathbf{q} + \mathbf{n})_l$, which is a noisy version of what we are interested in so that the scheme should be now analyzed within the framework of stochastic approximation [19]. The scheme is a building block for distributed implementation of various power control algorithms [19].

B. Computing General Functions

The discussion in the previous section shows that a network with K interfering links can be used to efficiently estimate weighted sums of some local variables/measurements. The idea can be extended to more general functions by letting the local variables be pre-processed prior to transmissions followed by a suitable receiver-side post-processing

of the received signal powers. To be more precise, let $\mathbf{x} = (X_1, \ldots, X_K) \in \mathbb{R}^K$ be a vector of some local variables and let $\varphi_k : \mathbb{R} \to \mathbb{R}_+, 1 \leq k \leq K$, be given pre-processing functions so that the kth transmitter sends a sequence of independent zero-mean random symbols at power $\varphi_k(X_k)/d_k$ where $d_k > 0$ is a constant that may be chosen subject to power constraints. The lth receiver estimates the average signal power y_l (by for instance averaging over all symbol intervals) and computes $\psi_l(y_l)$ where $\psi_l : \mathbb{R}_+ \to \mathbb{R}$ is a given post-processing function. Now, if $\mathbf{G} = (V_{k,l})_{1 \leq k,l \leq K} \in \mathbb{R}_+^{K \times K}$ and $\mathbf{D} = \mathrm{diag}(1/d_1, \ldots, 1/d_K)$, then it may be verified that[2]

$$\psi(\mathbf{y}) \approx \psi(\mathbf{GD}\varphi(\mathbf{x})) \qquad (3)$$

where $\psi(\mathbf{y}) = (\psi_1(y_1), \ldots \psi_K(y_k))$, $\varphi(\mathbf{x}) = (\varphi_1(x_1), \ldots, \varphi_K(x_K))$ and where we have neglected the receiver noise for brevity reasons.

The left-hand side of (3) is used to compute the estimate $\hat{f}(X_1, \ldots, X_K)$ of $f(X_1, \ldots, X_K)$, while pre-processing and post-processing are chosen to minimize the error by matching the resulting channel to the function as close as possible. The basic rationale behind this approach is the following result.

Theorem 2 ([23]). For every function f (from any compact metric space into the set of real numbers), there exist pre-processing functions φ_k and a post-processing function ψ with

$$f(x_1, \ldots, x_K) = \psi\left(\sum\nolimits_{k=1}^K \varphi_k(x_k)\right). \qquad (4)$$

The function space defined by (4) is in mathematics known as the space of *nomographic functions* (see [23] for details and examples), which owe their name from graphical representations (i.e., nomographs) that were formerly often used in engineering for solving certain types of nonlinear equations. A nice property of nomographic functions is that the pre-processing functions can be chosen to be universal in the sense that they do not depend on f [24], which implies that they do not need to be updated if the function to be computed changes.

It is pointed out that the pre-processing functions in (4) are not necessarily continuous. If the continuity property is necessary, then any function can be represented as a sum of not more than $2K + 1$ nomographic functions, which in practice means additional wireless resources. However, despite the additional resources, huge performance gains are possible in more general networks consisting of clusters of nodes [25].

In addition to the problem of noisy measurements and finite sequence lengths, there are two basic challenges associated with this approach. First, if the receivers are interested in different functions, then there is a need for universal pre-processing functions that can be used to compute any function of interest. In other words, it is desired that for given pre-processing functions, the function to be computed at the receivers only depend on the choice of the post-processing functions. The existence of universal pre-processing functions has been shown in [24]. Secondly, the channel inversions as

---

[2]The approximation means that the right-hand side is arbitrarily close to the left-hand side provided that the random sequences are sufficiently long.

described above do not eliminate the fading effects so that the received power at each receiver is a weighted sum of pre-processed variables/measurements; moreover the weights are unknown. Under certain conditions on fading distributions, this is not a problem since the superposition property of the wireless channel helps to average out the fading effects [26]. Moreover, multiple antenna systems can also be used to eliminate the fading effects [26]; these effects can be completely removed in the case of wireless multiple-access channel as discussed in the following section.

C. Computation via Multiple-Access Channels (CoMAC)

A special case of the network described in the previous section is a wireless multiple-access channel in which a *function* $f(X_1, \ldots, X_K)$ is to be computed at a single designated receiver called fusion center. In this case, denoting the fusion center as the first receiver,[3] the impact of the wireless channel can be eliminated by the channel inversions with $d_{k,k} = \alpha V_{1,k}$, where $\alpha > 0$ is chosen to satisfy some power constraints. In what follows, we assume $\alpha = 1$ for brevity and conclude from (3) that the estimate of the function f is given by

$$\hat{f}(X_1, \ldots, X_K) = \psi\left(\sum_{k=1}^{K} \varphi_k(X_k) + n\right), \qquad (5)$$

where $n \in \mathbb{R}_+$ denotes the instantaneous overall noise power, consisting of the power of all noise sources [27]. The argument in (5) describes the affine model of a wireless multiple-access channel, which is in its core the *superposition* of the pre-processed sensor readings and therefore a noisy version of the desired function $f(X_1, \ldots, X_K) = \psi\left(\sum_k \varphi_k(X_k)\right)$.

Figure 1 depicts a simulation example for a network consisting of $K = 25$ nodes. The system operating points (i.e., mean signal-to-noise ratios (SNR)) vary in a range that is typical for sensor network applications. We compare the analog CoMAC approach (i.e., the K nodes transmit simultaneously) with a standard TDMA in which sensor readings are quantized and separately transmitted as uncoded bit streams. In doing so, the system performance is measured in terms of the estimation accuracy, which is defined to be the absolute value of the deviation between the estimate $\hat{f}$ and the actual desired function f, relative to the range (i.e., $|E| := |\frac{\hat{f}-f}{f_{\max}-f_{\min}}|$).

III. Applications

The analog computation scheme presented here is a fundamental building block of distributed in-network processing for numerous applications. Examples are the fast converging cluster-based consensus algorithms presented in [28], [29] and the digital counterpart from [30]. In the remainder, we briefly discuss other applications.

A. Distributed Optimization and Distributed Learning

A vast number of problems in, among other fields, multi-agent systems, distributed control, distributed machine learning, and distributed signal processing can be posed as a convex

---

[3]In the context of wireless networks with K links, this means that there is one receiver and the K desired power gains from the transmitters to the receiver are $V_{1,k}, 1 \leq k \leq K$.
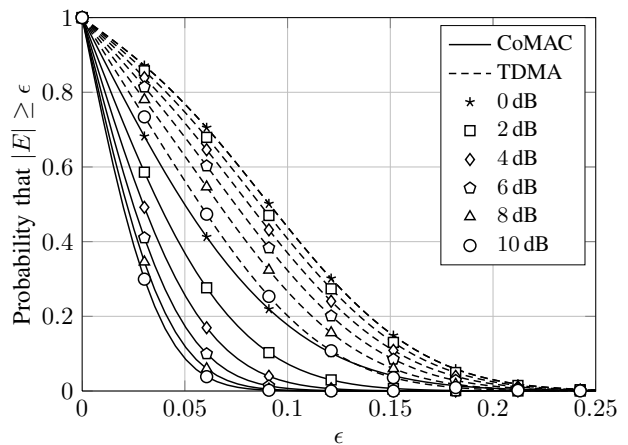
Fig. 1: CoMAC vs. TDMA for $K = 25$ sensor nodes, quantization with 10 bit, SNR $= 0, 2, 4, 6, 8, 10$ dB and the function "arithmetic mean". CoMAC (solid lines) significantly outperforms TDMA (dashed lines) for all SNR.

optimization problem where the objective is to minimize a global function that is defined as the sum of local functions of an agent in the network [31]–[34]. Each local function is typically private information of an agent, and it is constructed with local information. Examples of applications include acoustic source localization [34], coordination of unmanned vehicles [31], environmental modeling [34], and distributed adaptive filtering [32]. Common to these applications is the fact that they are variations of the following two-step approach. In the first step, agents apply an iteration of a subgradient method to their local functions in order to improve their individual estimates of a minimizer of the global function. In the second step, agents fuse their improved estimates with those of their local neighbors by computing local weighted averages. As a result, efficient distributed computation of weighted averages is of paramount importance to this two-step approach, and we note that weighted averages can be efficiently computed with the solutions described in the previous section.

B. Principal Subspace Tracking and Eigen-based Detection

The CoMAC paradigm can be exploited in many applications that involve distributed estimation or tracking of the "principal subspace" (i.e., the principal eigenvalues and eigenvectors of a sample covariance matrix). Such applications include multi-sensor array detection and estimation [35], distributed traffic anomaly detection [36]–[38], distributed eigenvalue-based spectrum sensing in cognitive radio networks [39], etc. Many of the algorithms heavily rely on average consensus. Therefore, by replacing the classic average consensus approach by the CoMAC protocol, we expect huge performance gains since classic consensus algorithms require communication protocols to schedule the exchange of messages among nodes. Using CoMAC, if users are coarsely synchronized, only little coordination is needed.

Finally, we remark that in certain detection problems (for instance, eigenvalue-based spectrum sensing in cognitive radios [40]) it is more important to estimate eigenvalues $\lambda_1, \ldots, \lambda_K$ once N samples are received (at the end of the sensing period)

rather than updating the estimates at every new sample. In this scenario, numerical methods such as the Lanczos algorithm [41] are more suitable than the well known Oja-Karhunen recursion [42]. In [39] we have shown that the Lanczos algorithm can be reformulated as a sequence of steps that involve either average consensus or internal computations, thus leading to decentralized eigenvalue-based signal detection. Also in this case, the analog CoMAC approach can be used to perform average consensus operations.

REFERENCES

[1] A. Giridhar and P. R. Kumar, "Toward a theory of in-network computation in wireless sensor networks," *IEEE Commun. Mag.*, vol. 44, no. 4, pp. 98–107, Apr. 2006.

[2] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3498–3516, Oct. 2007.

[3] M. Gastpar and M. Vetterli, "Source-channel communication in sensor networks," in *Proc. 2nd Int. Workshop Information Processing in Sensor Networks (IPSN)*, Palo Alto, CA, USA, Apr. 2003, pp. 162–177.

[4] M. Gastpar, M. Vetterli, and L. Dragotti, "Sensing reality and communicating bits: A dangerous liaison," *IEEE Signal Process. Mag.*, vol. 23, no. 4, pp. 70–83, Jul. 2006.

[5] C. E. Shannon, "A mathematical theory of communications," *Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[6] G. Mergen and L. Tong, "Type based estimation over multiaccess channels," *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.

[7] W. U. Bajwa, J. D. Haupt, A. M. Sayeed, and R. D. Nowak, "Joint source-channel communication for distributed estimation in sensor networks," *IEEE Trans. Inf. Theory* vol. 53, no. 10, pp. 3629–3653, Oct. 2007.

[8] M. K. Banavar, C. Tepedelendioğlu, and A. Spanias, "Estimation over fading channels with limited feedback using distributed sensing," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 414–425, Jan. 2010.

[9] W. Li and H. Dai, "Distributed detection in wireless sensor networks using a multiple access channel," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 822–833, Mar. 2007.

[10] G. Mergen, V. Naware, and L. Tong, "Asymptotic detection performance of type-based multiple access over multiaccess fading channels," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 1081–1092, Mar. 2007.

[11] K. Liu and A. M. Sayeed, "Type-based decentralized detection in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 1899–1910, May 2007.

[12] S. Zhang, S.-C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *Proc. 12th Ann. Int. Conf. Mobile Computing and Networking (ACM MobiCom)*, Los Angeles, CA, USA, Sep. 2006, pp. 358–365.

[13] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: Analog network coding," in *Proc. Conf. Applications, Technologies, Architectures, and Protocols for Computer Communications (ACM SIGCOMM)*, Kyoto, Japan, Aug. 2007, pp. 397–408.

[14] B. Nazer and M. Gastpar, "Reliable physical layer network coding," *Proc. IEEE*, vol. 99, no. 3, pp. 438–460, Mar 2011.

[15] ——, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.

[16] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory* vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[17] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[18] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.

[19] S. Stańczak, M. Wiczanowski, and H. Boche, *Fundamentals of Resource Allocation in Wireless Networks*, ser. Foundations in Signal Processing, Communications and Networking. Berlin: Springer, 2009, vol. 3.

[20] S. Stańczak and M. Wiczanowski, "Distributed fair power control for wireless networks: Objectives and algorithms," in *Proc. 43rd Ann. Allerton Conf. Communications, Control, and Computing*, Monticello, IL, USA, Sept. 2005.

[21] S. Stańczak, M. Wiczanowski, and H. Boche, "Distributed power control for optimizing a weighted sum of link-layer QoS values," in *Proc. IEEE Global Telecommunications Conf. (GLOBECOM)*, St. Louis, MO, USA, Nov. 2005, pp. 3539–3543.

[22] ——, "Distributed utility-based power control: Objectives and algorithms," *IEEE Trans. Signal Process.*, vol. 55, no. 10, pp. 5058–5068, Oct. 2007.

[23] M. Goldenbaum, H. Boche, and S. Stańczak, "Analyzing the space of functions analog-computable via wireless multiple-access channels," in *Proc. 8th Int. Symp. Wireless Communication Systems (ISWCS)*, Aachen, Germany, Nov. 2011, pp. 779–783.

[24] ——, "Analog computation via wireless multiple-access channels: Universality and robustness," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, Apr. 2012, pp. 2921–2924.

[25] ——, "On analog computation of vector-valued functions in clustered wireless sensor networks," in *Proc. 46th Annu. Conf. Information Sciences and Systems (CISS)*, Princeton, NJ, USA, Mar. 2012.

[26] M. Goldenbaum and S. Stańczak, "Computing functions via SIMO multiple-access channels: How much channel knowledge is needed?," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Dallas, TX, USA, Mar. 2010, pp. 3394–3397.

[27] M. Goldenbaum, S. Stańczak, and M. Kaliszan, "On function computation via wireless sensor multiple-access channels," in *Proc. IEEE Wireless Communications & Networking Conf. (WCNC)*, Budapest, Hungary, Apr. 2009.

[28] M. Zheng, M. Goldenbaum, S. Stańczak, and H. Yu, "Fast average consensus in clustered wireless sensor networks by superposition gossiping," in *Proc. IEEE Wireless Communications & Networking Conf. (WCNC)*, Paris, France, Apr. 2012, pp. 2009–2014.

[29] M. Goldenbaum, H. Boche, and S. Stańczak, "Nomographic gossiping for f-consensus," in *Proc. 10th Int. Symp. Modeling and Optimization in Mobile, Ad-Hoc, and Wireless Networks (WiOpt)*, Paderborn, Germany, May 2012, pp. 130–137.

[30] B. Nazer, A. G. Dimakis, and M. Gastpar, "Local interference can accelerate gossip algorithms," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 876–887, Aug. 2011.

[31] B. Johansson, A. Speranzon, M. Johansson, and K. H. Johansson, "On decentralized negotiation of optimal consensus," *Automatica*, vol. 44, no. 4, pp. 1175–1179, Apr. 2008.

[32] R. L. G. Cavalcante, I. Yamada, and B. Mulgrew, "An adaptive projected subgradient approach to learning in diffusion networks," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2762–2774, Jul. 2009.

[33] K. Srivastava and A. Nedic, "Distributed asynchronous constrained stochastic optimization," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 772–790, Aug. 2011.

[34] R. L. G. Cavalcante, A. Rogers, N. R. Jennings, and I. Yamada, "Distributed asymptotic minimization of sequences of convex functions by a broadcast adaptive subgradient method," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 739–753, Aug. 2011.

[35] L. Li, A. Scaglione, and J. H. Manton, "Distributed principal subspace estimation in wireless sensor networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 725–738, Aug. 2011.

[36] L. Huang, X. L. Nguyen, M. Garofalakis, M. I. Jordan, A. Joseph, and N. Taft, "In-network PCA and anomaly detection," *Advances in Neural Information Processing Systems 19*, MIT Press, 2007, pp. 617–624.

[37] T. Ide and H. Kashima, "Eigenspace-based anomaly detection in computer systems," in *Proc. 10th Int. Conf. Knowledge Discovery and Data Mining (ACM KDD)*, Seattle, WA, USA 2004, pp. 440–449.

[38] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," in *Proc. Conf. Applications, Technologies, Architectures, and Protocols for Computer Communications (ACM SIGCOMM)*, Portland, OR, USA, 2004, pp. 219–230.

[39] F. Penna and S. Stańczak, "Decentralized largest eigenvalue test for multi-sensor signal detection", submitted to *IEEE Global Communications Conf. (GLOBECOM)*, 2012.

[40] B. Nadler, F. Penna, and R. Garello, "Performance of eigenvalue-based signal detectors with known and unknown noise level," in *Proc. IEEE Int. Conf. Communications (ICC)*, Kyoto, Japan, June 2011.

[41] Y. Saad, *Iterative methods for sparse linear systems*, 2nd ed., SIAM, 2003.

[42] E. Oja and J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix," *J. Math. Anal. Appl.*, vol. 106, no. 1, pp. 69–84, Feb. 1985.