Distributed Projection Approximation Subspace Tracking Based on Consensus Propagation

Carolina Reyes
Inst. of Communications and RF Engineering
Vienna University of Technology
1040 Vienna, Austria
creyes@nt.tuwien.ac.at

Thibault Hilaire
Laboratory of Computer Science (LIP6)
University Pierre and Marie Curie
Paris, France
thibault.hilaire@lip6.fr

Christoph F. Mecklenbräuker
Christian Doppler Laboratory for
Wireless Technologies for Sustainable Mobility
Vienna University of Technology
1040 Vienna, Austria
cfm@nt.tuwien.ac.at

***Abstract*—We develop and investigate a distributed algorithm for signal subspace tracking with a wireless sensor network without the need for a fusion center, in order to improve the robustness and scalability. We assume that all sensor nodes may broadcast messages to sensors in their neighborhood defined by a finite (small) communication radius.**

To this aim, we start from Projection Approximation Subspace Tracking (PAST) which is a well-investigated algorithm suitable for implementation in a fusion center. We arrive at a distributed approximation of the PAST algorithm by letting each sensor broadcast its local observation variable $x_n(t)$ and a filtered observation vector $\underline{\boldsymbol{y}}_n(t)$ to its neighborhood. Vice versa, the received messages at sensor node n from its neighborhood are fused by employing consensus propagation.

Finally, we investigate the proposed distributed algorithm in simulation runs.

I. Introduction

Wireless sensor networks consist of a number of nodes which sense environmental changes and report to other nodes inside the network. There are many applications ranging from industrial, building, home system automation to monitoring of chemicals in hydrology, agriculture, and pollution control, as well as the prediction of avalanches and land slides.

In many applications, aggregate functions of the sensor data are more important than individual node data. When the scale of the wireless sensor network is large, the average value collected from the whole network is more important than the value collected by one single node, as stated in [1], [2]. The relevant subspace in which the sensor data are confined is an important aggregate statistic which is a pre-requisite for many types of data compression techniques, signal detection, classification, and localization.

Sensor nodes are adequate for deployment in harsh environments or over large geographical areas, however most of the state-of-the-art sensor network architectures are rather centralized. The resulting drawbacks motivate us to the current research:

- Since the sensor nodes might be located in remote areas, the use of batteries for powering is necessary. However, when the lifetime of these batteries is used up, then it is very difficult to replace them. Due to this fact, it is a major concern to decrease the power consumption as much as possible.
- The communication channel constrains the data rate for message transmission.

We concentrate our efforts on distributing the Projection Approximation Subspace Tracking (PAST) algorithm [3] with a low amount of message passing among sensor nodes. The latter is motivated by the high power consumption for transmitting messages.

Notation: We denote a column vectors in underlined boldface and matrices in uppercase and boldface. The superscript ^{H} represents complex conjugation, $^\top$ transposition, $tr(.)$ the trace operator, $E[.]$ the expectation and $\|.\|$ the Euclidean vector norm.

Organization of the paper: Section II introduces the PAST algorithm, the analytical model and corresponding constraints. The section III presents a survey on distributed averaging as an alternative for decentralization. The fusion of both PAST and consensus propagation methods in order to obtain a distributed version of the original algorithm is presented in Section IV. Finally, Section V shows and discusses specific simulation results which highlight the tracking capabilities of our distributed approach.

II. Projection Approximation Subspace Tracking

In the recent years, *subspace methods* have become a very researched and remarked method in the realm of modern signal processing. Some of its applications encompasses blind communication channel identification [4], estimation of direction of arrivals from signals impinging on an antenna array and image compression, to mention some. Methods based on the singular value decomposition of the sample covariance matrix for every time step lead to a high computational burden, cf. [5], [6]. The PAST algorithm does not require a sample covariance estimate. It estimates the signal subspace at time t recursively, depending on the previous subspace estimate at time $t-1$ and the new observation $\underline{\boldsymbol{x}}(t)$. Yang [3] introduces a new signal subspace model interpretation, considering an unconstrained minimization function. The subspace tracking is implemented by employing the recursive least squares algorithm, based on an appropriate projection approximation.

A. Mathematical model

Let $\underline{\boldsymbol{x}}(t) \in \mathbb{C}^N$ be the data vector observed at time t, composed of r narrow-band signal waves impinging on a planar array of sensors hidden in additive noise (zero-mean with variance σ^2). The data vector is modelled as

$$\underline{x}(t) = \boldsymbol{A}(\underline{\omega})\underline{s}(t) + \underline{v}(t) \ , \tag{1}$$

$$\boldsymbol{A}(\underline{\omega}) = \begin{pmatrix}\underline{a}(\omega_1) & \underline{a}(\omega_2) & \dots & \underline{a}(\omega_r)\end{pmatrix} , \tag{2}$$

$$\underline{s}(t) = \begin{pmatrix} s_1(t) \\ \vdots \\ s_r(t) \end{pmatrix} , \quad \underline{v}(t) = \begin{pmatrix} v_1(t) \\ \vdots \\ v_N(t) \end{pmatrix} . \tag{3}$$

$\boldsymbol{A}(\underline{\omega})$ is a deterministic $N \times r$ matrix depending on $\underline{\omega} = (\omega_1, \dots, \omega_r)$ whose columns are plane wave steering vectors $\underline{a}(\omega_i)$. The n-th element of $\underline{a}(\omega)$ is

$$[\underline{a}(\omega_i)]_n = \exp\left(\mathrm{j}\frac{2\pi}{\lambda}(\xi_n \cos\theta_i + \eta_n \sin\theta_i)\right) / \sqrt{N}$$

where frequency $\omega_i = \cos\theta_i$. The Cartesian coordinates of the n-th sensor node are (ξ_n, η_n) and λ is the wavelength. N is the number of sensors and r the number of impinging waves. $\underline{s}(t)$ is the signal vector and $\underline{v}(t)$ is uncorrelated additive noise.

We allow $\underline{\omega}$ to be slowly time-varying, i.e. $\underline{\omega} = \underline{\omega}(t)$ and seek an estimate for matrix $\boldsymbol{W}(t)$ whose columns span the same subspace as $\boldsymbol{A}(\underline{\omega}(t))$. A *tracking algorithm* estimates $\boldsymbol{W}(t)$ by a function of $\boldsymbol{W}(t-1)$ and $\underline{x}(t)$ alone.

B. Brief review of PAST

Yang [3] approximates the cost function to minimize

$$J(\boldsymbol{W}(t)) = \sum_{i=1}^{t} \beta^{t-i} \left\| \underline{x}(i) - \boldsymbol{W}(t)\boldsymbol{W}^H(t)\underline{x}(i) \right\|^2 \tag{4}$$

by

$$J'(\boldsymbol{W}(t)) = \sum_{i=1}^{t} \beta^{t-i} \left\| \underline{x}(i) - \boldsymbol{W}(t)\underline{y}(i) \right\|^2 \tag{5}$$

with

$$\underline{y}(i) = \boldsymbol{W}^H(i-1)\underline{x}(i) \ . \tag{6}$$

Here, the matrix $\boldsymbol{W} \in \mathbb{C}^{N \times r}$ is constrained to rank $r < N$.

The minimization of cost function (5) results in a low-complexity update of the signal subspace. The computational complexity is of order $3Nr + O(r^2)$ operations per time step, where N is the number of sensors and r the number of tracked signals.

In **Algorithm 1**, $\underline{y}(t)$ stores the modified data vector resulting from the multiplication between the old signal subspace $\boldsymbol{W}^H(t-1)$ and the new data vector $\underline{x}(t)$ observed at the sensor array. The following computational steps in **Algorithm 1** are derived from the Recursive Least Square algorithm [7] to track the signal subspace matrix $\boldsymbol{W}(t)$.

III. Distributed Averaging Algorithms

The PAST algorithm is structured in a centralized way, and we aim to distribute it for sensor network applications. Namely, we wish to achieve two important goals for large scale distributed systems: (i) Fault tolerance: consider large scale systems that are prone to communication disruptions due to link/node failures (Robustness). (ii) Simplicity and scalability: pursuit of a simple topology network able to adapt itself according to the volatility of the system. The use of a centralized network limits the scalability of the system.

Algorithm 1: PAST algorithm by Yang [3]

Input: β, $\boldsymbol{P}(0)$, $\boldsymbol{W}(0)$
for $t := 1, 2, \dots$ **do**
  Input: $\underline{x}(t)$
  $\underline{y}(t) = \boldsymbol{W}^H(t-1)\underline{x}(t)$
  $\underline{h}(t) = \boldsymbol{P}(t-1)\underline{y}(t)$
  $\underline{g}(t) = \underline{h}(t)/[\beta + \underline{y}^H(t)\underline{h}(t)]$
  $\boldsymbol{P}(t) = \frac{1}{\beta}[\boldsymbol{P}(t-1) - \underline{g}(t)\underline{h}^H(t)]$
  $\underline{e}(t) = \underline{x}(t) - \boldsymbol{W}(t-1)\underline{y}(t)$
  $\boldsymbol{W}(t) = \boldsymbol{W}(t-1) + \underline{e}(t)\underline{g}^H(t)$
  Output: $\boldsymbol{W}(t)$
endfor

In recent work, distributed adaptive algorithms have been proposed to address the issue of estimation over distributed networks. These new algorithms outperform the classical centralized solution, but are based on specific network topologies which lead to scalability constrains. For example, [8] considers the design of distributed architectures based on randomized graph models.

Different strategies are used for distributed fusion and average of the information. Some algorithms are based on interaction between a node and all its adjacent nodes in the network, so as to reach a *consensus*, namely the Consensus Propagation algorithms, introduced by Olfati-Saber and Murray in [9], [10].

Another strategy is based on pairwise communications: the gossip-based algorithms such as the push-sum protocol introduced by Kempe in [1] are good alternatives when low communication overhead is desired.

In this paper, we are using a vector version of the Consensus Propagation, with constant weights (see [11] for the general case, with non-constant weights), as presented in **Algorithm 2**.

For every node $n = 1, \dots, N$ in the network, $\mathcal{N}_n$ denotes the set of the adjacent nodes, including itself. We assume that every node in the *neighborhood* $\mathcal{N}_n$ can correctly receive broadcasted messages from node n with probability 1, cf. Figure 1.

At the end of step $t-1$, every node n sends to its adjacent nodes $\mathcal{N}_n$ its own estimation of the average $\underline{y}_n(t-1)$ and a weight w_n. Then, at the beginning of step t, every node n receives the pairs $\{\underline{y}_i(t-1), w_i\}_{i \in \mathcal{N}_n}$ from its neighbors and compute a new average, to be sent at the end of step t.
The constant weights are chosen as

$$w_n = 1/\sqrt{|\mathcal{N}_n|} \ . \tag{7}$$

Algorithm 2 can be rewritten as follows. Let $\boldsymbol{Y}(t) \in \mathbb{C}^{N \times r}$ be a matrix aggregating all $\underline{y}_n(t)$

$$\boldsymbol{Y}(t) \triangleq \begin{pmatrix}\underline{y}_1(t)^\top & \dots & \underline{y}_N(t)^\top\end{pmatrix}^\top \ . \tag{9}$$

Algorithm 2: Consensus Propagation Algorithm

for $n := 1, 2, \dots, N$ **do**

Input: $\{\underline{\boldsymbol{y}}_j(t-1), w_j\}_{j\in\mathcal{N}_n}$ are the pairs sent to node n in step $t-1$

$$\underline{\boldsymbol{y}}_n(t) = \left(\sum_{j\in\mathcal{N}_n} \underline{\boldsymbol{y}}_j(t-1)w_j\right) \Big/ \left(\sum_{j\in\mathcal{N}_n} w_j\right) \quad (8)$$

Broadcast the pair $\{\underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes in $\mathcal{N}_n$

Output: $\underline{\boldsymbol{y}}_n(t)$ is the estimation of the average in step t at node n

endfor

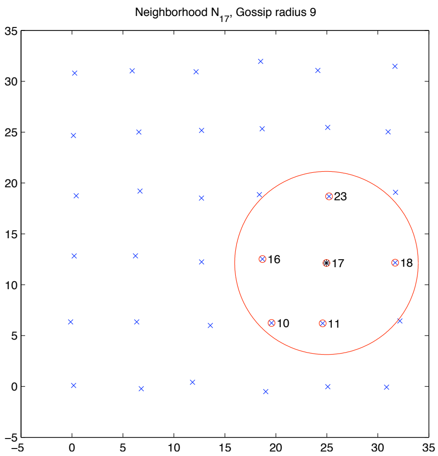

Fig. 1. Sensor network with neighborhood $\mathcal{N}_{17}$ for radius 9.

Matrix $\boldsymbol{Y}(t)$ is evolving according to $\boldsymbol{Y}(t+1) = \boldsymbol{\Omega}\boldsymbol{Y}(t)$ where

$$\boldsymbol{\Omega} = \left(\text{diag}\left(\mathcal{A}\underline{\boldsymbol{w}}^\top\right)\right)^{-1} \mathcal{A}\ \text{diag}(\underline{\boldsymbol{w}}), \quad (10)$$

$\underline{\boldsymbol{w}} = (w_1, \dots, w_N)$ and $\mathcal{A}$ is the adjacency matrix of the undirected graph ($\mathcal{A}_{i,j} = 1$ if nodes i and j can communicate). The speed of the convergence is given by the second largest eigenvalue of $\boldsymbol{\Omega}$ [2].

IV. PAST-Consensus Propagation Algorithm

Recall the centralized **Algorithm 1**, that is employed within each node of the system for distributing the computations, with minor differences that are explained below. First, consider a wireless sensor network with a planar square array structure and a number of nodes N. Every node n observes (environmental) data $x_n(t)$, assumed to be scalar, that are shared with all nodes $j \in \mathcal{N}_n$ inside a specific broadcasting neighborhood. This means that each node n receives the observations from its adjacent nodes $x_j(t-1)$, that are used for aggregating the local observation vector $\underline{\boldsymbol{x}}_n(t)$. We define the local observation vector $\underline{\boldsymbol{x}}_n(t)$ as the aggregation of $\{x_j(t-1)\}_{j\in\mathcal{N}_n}$, i.e.

$$\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n \underline{\boldsymbol{x}}(t-1) \quad (11)$$

where the $|\mathcal{N}_n| \times N$ selection matrix $\boldsymbol{S}_n$ is defined by

$$(\boldsymbol{S}_n)_{ij} = \begin{cases} 1 & \text{if } j = i\text{-th node } \in \mathcal{N}_n \\ 0 & \end{cases}, \quad (12)$$

and $\underline{\boldsymbol{x}}(t)$ is the data vector observed in the whole network.

This information exchange permits that all nodes involved in the communication process have knowledge of the neighboring observations. We propose to locally average the vector $\underline{\boldsymbol{y}}_n(t)$ in node n by fusing information aggregated in its associated neighborhood $\mathcal{N}_n$. The local averaging is described by Eq.(8) within **Algorithm 2**. Here, we calculate the internal $\underline{\boldsymbol{y}}_n(t)$ taking into account the weighted $\underline{\boldsymbol{y}}_j(t-1)$. This is subsequently divided by the sum of the accumulated weight of w_n and the neighboring weights w_j. Thus, a distributed average version of $\underline{\boldsymbol{y}}_n(t)$ is assured by means of the Consensus Propagation algorithm.

After defining in which part of the algorithm the distribution actually takes place, we introduce the **Algorithm 3**. The local vector $\underline{\boldsymbol{y}}_n(t)$ is calculated at every node of the system using the **Algorithm 2** and the following calculations follow the original **Algorithm 1** except for the diagonal weighting matrix $\boldsymbol{D}_n = \boldsymbol{S}_n\text{diag}(\underline{\boldsymbol{w}})\boldsymbol{S}_n^\top$. The reason why we decided to distribute this particular variable is because according to (6), this vector contains information from the updated signal subspace at $t-1$ as well as new arriving observation data x_n and x_j. As a result, every sensor in the system has indirect knowledge of these parameters.

Algorithm 3: PAST-Consensus

Input: β, $\boldsymbol{P}_1(0), \dots, \boldsymbol{P}_N(0)$, $\boldsymbol{W}_1(0), \dots, \boldsymbol{W}_N(0)$

for $t := 1, 2, \dots$ **do**

  for $n := 1, 2, \dots, N$ **do**

    Input: $x(n)$

    aggregate $\underline{\boldsymbol{x}}_n(t) = \boldsymbol{S}_n\underline{\boldsymbol{x}}(t-1)$ from all nodes$\in\mathcal{N}_n$

    $\underline{\boldsymbol{y}}_n(t) = \boldsymbol{W}_n^H(t-1)\underline{\boldsymbol{x}}_n(t)$

    apply **Algorithm 2** for locally averaging $\underline{\boldsymbol{y}}_n(t)$

    $\underline{\boldsymbol{h}}_n(t) = \boldsymbol{P}_n(t-1)\underline{\boldsymbol{y}}_n(t)$

    $\underline{\boldsymbol{g}}_n(t) = \underline{\boldsymbol{h}}_n(t)/[\beta + \underline{\boldsymbol{y}}_n^H(t)\underline{\boldsymbol{h}}_n(t)]$

    $\boldsymbol{P}_n(t) = \frac{1}{\beta}[\boldsymbol{P}_n(t-1) - \underline{\boldsymbol{g}}_n(t)\underline{\boldsymbol{h}}_n^H(t)]$

    $\underline{\boldsymbol{e}}_n(t) = \boldsymbol{D}_n(\underline{\boldsymbol{x}}(t) - \boldsymbol{W}_n(t-1)\underline{\boldsymbol{y}}_n(t))$

    $\boldsymbol{W}_n(t) = \boldsymbol{W}_n(t-1) + \underline{\boldsymbol{e}}_n(t)\underline{\boldsymbol{g}}_n^H(t)$

    broadcast $\{x_n(t), \underline{\boldsymbol{y}}_n(t), w_n\}$ to all nodes$\in\mathcal{N}_n$

  endfor

endfor

V. Simulations

We simulate a planar sensor network with $N = 36$ nodes, cf. Figure 1. The locations of the nodes are chosen as an imperfect Cartesian grid. The average distance between nodes is $\lambda/2$ and the transmission range is $1.44\lambda/2$.

The sensor network observes 1000 snapshots in time. The sensor network tracks the instantaneous frequencies of $r = 2$ impinging waves which are linearly time-varying. The true instantaneous frequencies are shown in the figures below. The forgetting factor $\beta = 0.97$ is chosen. We initialize $\boldsymbol{W}(0)$ and $\boldsymbol{P}(0)$ to identity. We assume that messages sent by node n are received correctly by all nodes in neighborhood $\mathcal{N}_n$. For displaying the subspace tracking behavior vs. time, we apply

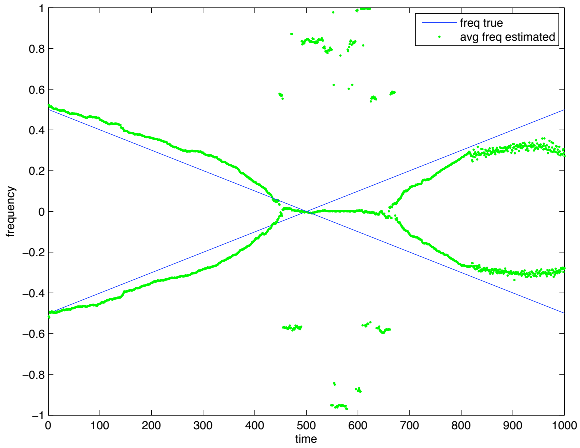

Fig. 2. PAST result for whole sensor array ($N = 36$, $r = 2$)

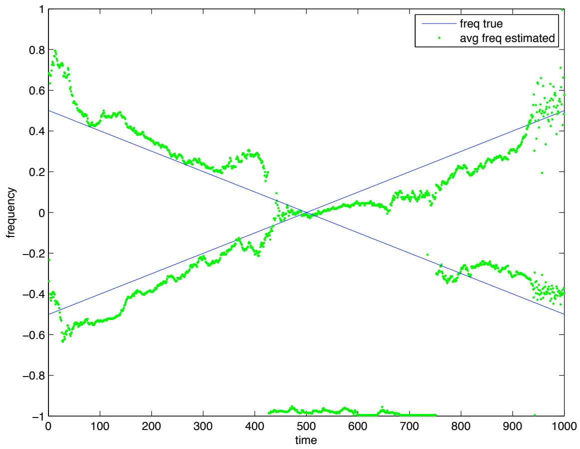

Fig. 3. PAST result neighborhood $\mathcal{N}_{17}$ ($N = 6$, $r = 2$)

the spectral MUSIC algorithm [6] for extracting r frequency components from the signal subspace matrices $\boldsymbol{W}(t)$ over time t. All simulation runs are carried out with one and the same noise realisation.

In **Figure** 2, we observe the tracking of the original PAST Algorithm 1. The true instantaneous signal frequencies are represented by blue lines. The green dots correspond to the tracked frequencies. Likewise, in **Figure** 3, we plot the tracking behavior of the original PAST algorithm for the subarray defined by the neighborhood $\mathcal{N}_{17}$ which contains 6 sensors. Finally, in **Figure** 4, we illustrate the tracking behavior of Algorithm 3. These preliminary results indicate that its behavior is closer to **Figure** 2 than **Figure** 3.

VI. Summary and Conclusion

PAST-Consensus (**Algorithm 3**) implements a distributed variant of the PAST algorithm. Every sensor node n employs the PAST algorithm (**Algorithm 1**) to *locally* track the signal subspace and uses **Algorithm 2** to couple its own observation and internal state vector with its neighborhood $\mathcal{N}_n$ of sensor nodes. To this aim, every sensor node n broadcasts its local observation $x_n(t)$, its locally filtered r-dimensional vector

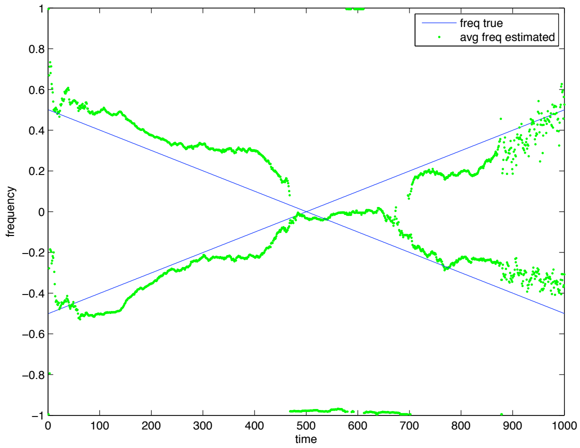

Fig. 4. Algorithm 3 result for sensor No. 17 ($r = 2$)

$\underline{\boldsymbol{y}}_n(t)$, and a weight w_n to its neighborhood $\mathcal{N}_n$ of sensor nodes. Thus, every node broadcasts $2(r+1)+1$ real-valued variables per time step. From these preliminary results, we believe that a well performing low-cost implementation of a distributed PAST algorithm with low communication overhead is a feasible goal.

Acknowledgment

Support was provided by the fFORTE-Women in Technology Program at Vienna Univ. of Technology, the Austrian Ministry for Science and Research, the Christian Doppler Lab *Wireless Technologies for Sustainable Mobility*, and the WWTF project *Distributed Information Processing for Spatio-Temporal Fields in Wireless Sensor Networks*. We thank Erwin Riegler (ftw) and the national research network on *Signal and Information Processing in Science and Engineering* (NFN-SISE) for many fruitful discussions.

References

[1] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Foundations of Computer Science*, 2003.
[2] R. Olfati-saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," in *Proceedings of the IEEE*, vol. 95, pp. 205–233, Jan. 2007.
[3] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Sig. Proc.*, vol. 43, no. 1, pp. 95–107, 1995.
[4] J. Akhtman and L. Hanzo, "Decision directed channel estimation employing projection approximation subspace tracking," *IEEE Proceedings*, pp. 3056–3060, 2007.
[5] R. Kumaresan and D. W. Tufts, "Estimating the angles of arrival of multiple plane waves," *IEEE Trans. on Aerospace and Electronic Systems*, pp. 134–139, 1983.
[6] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Ant. Prop.*, vol. AP-34, pp. 276 – 280, Mar. 1986.
[7] S. Haykin, *Adaptive Filter Theory*. Englewood-Cliffs (NJ), USA: Prentice-Hall, 1991.
[8] J. Kleinberg and E. Tardos, *Algorithm Design*. Pearson Int., 2006.
[9] R. Olfati-saber and R. Murray, "Consensus protocols for networks of dynamic agents," *American Control Conf.*, vol. 2, pp. 951–956, June 2003.
[10] R. Olfati-saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Contr.*, vol. 49, pp. 1520–1533, Sept. 2004.
[11] C. Moallemi and B. V. Roy, "Consensus propagation," *IEEE Trans. Inf. Theor.*, vol. 52, pp. 4753–4766, Nov. 2006.