

LEARNING BY NETWORKED AGENTS UNDER PARTIAL INFORMATION

Chung-Kai Yu and Ali H. Sayed

Department of Electrical Engineering
University of California, Los Angeles

ABSTRACT

In many scenarios of interest, agents may only have access to partial information about an unknown model or target vector. Each agent may be sensing only a subset of the entries of a global target vector, and the number of these entries can be different across the agents. If each of the agents were to solve an inference task independently of the other agents, then they would not benefit from neighboring agents that may be sensing similar entries. This work develops cooperative distributed techniques that enable agents to cooperate even when their interactions are limited to exchanging estimates of select few entries. In the proposed strategies, agents are only required to share estimates of their common entries, which results in a significant reduction in communication overhead. Simulations show that the proposed approach improves both the performance of individual agents and the entire network through cooperation.

Index Terms— Adaptive network, constrained optimization, penalty function, diffusion strategy, distributed estimation.

1. INTRODUCTION AND RELATED WORK

Adaptive networks enable agents to share information and to solve distributed optimization and inference tasks in an online and decentralized manner. In most prior works, agents are assumed to have a common minimizer and cooperate to estimate it by using effective distributed strategies such as the consensus strategy (e.g., [1–8]) or the diffusion strategy (e.g., [9–12]). When the agents do not share a minimizer, it was shown in [9, 10, 13, 14] that the network converges to a Pareto optimal solution. When it is desired instead that agents, or clusters of agents within the network, should converge to their respective models, rather than the Pareto solution, then multi-task diffusion strategies become useful and can be used to attain this objective [15, 16].

In these earlier contributions, it is generally assumed that the target vector for each agent has the same size and, moreover, that the agents sense data that is affected by all entries of their target vectors. In this work, we relax these conditions and consider a broader scenario where individual agents sense only a subset of the entries of the global target vector, and where different agents can sense subsets of different sizes. This formulation allows us to model the important situation in which agents may only have access to partial infor-

mation about an unknown model or target vector. If each of the agents were to solve an inference task independently of the other agents, then they would not benefit from cooperation with neighboring agents that may be sensing common entries. This work develops cooperative distributed techniques that enable agents to cooperate even when their interactions are limited to exchanging estimates of select few entries. To attain this objective, we allow for some entries of the global target vector to be observable by more than one agent so that cooperation across the network is justified.

Our approach will be based on formulating a constrained optimization problem for recovering partial entries of the global target vector. However, rather than solve this problem directly, we will introduce a penalized version using a quadratic term to penalize the violation of the constraints. We will then develop a diffusion learning solution to solve the optimization problem in a distributed manner by relying on two incremental steps. In the adaptation step, agents descend along the negative direction of the gradients of their costs. And in the penalty step, they descend along the negative direction of the gradients of their penalties. When the exact gradient information is unavailable, the observed data is used to compute instantaneous approximations for the gradient vectors. In the penalty step, agents will only share the common entries of their estimates with neighbors to reduce the communication costs. The order of executing the two incremental steps results in the Adapt-then-Penalize (ATP) or Penalize-then-Adapt (PTA) diffusion strategies.

We remark that this work considers a more general scenario than the partial diffusion formulation proposed in [17]. There, all agents sense data driven by the *same* target vector, cooperate to estimate this target vector, and exchange only part of their entries. In our formulation, each agent will be sensing data driven by different local target vectors and these can be of different sizes. In this way, agents are able to cooperate even if their target vectors are only partially common. We then show that sufficiently small step-sizes ensure mean and mean-square stability. We illustrate the results by means of computer simulations.

Notation: We use lowercase letters to denote vectors and scalars, uppercase letters for matrices, plain letters for deterministic variables, and boldface letters for random variables.

2. PROBLEM FORMULATION

At each time instant $i \geq 0$, each agent k is assumed to have access to a scalar measurement $\mathbf{d}_k(i) \in \mathbb{R}$ and a regression vector $\mathbf{u}_{k,i} \in \mathbb{R}^{1 \times M_k}$ with covariance matrix $R_{u,k} =$

This work was supported in part by NSF grants ECCS-1407712 and CCF-1524250. Emails: ckyuna@ucla.edu, sayed@ucla.edu.

$\mathbb{E} \mathbf{u}_{k,i}^\top \mathbf{u}_{k,i} > 0$. The regressors $\{\mathbf{u}_{k,i}\}$ are assumed to have zero mean and to be temporally white and spatially independent. The data $\{\mathbf{d}_k(i), \mathbf{u}_{k,i}\}$ are assumed to be related via the linear regression model:

$$\mathbf{d}_k(i) = \mathbf{u}_{k,i} w_k^o + \mathbf{v}_k(i) \quad (1)$$

where $w_k^o \in \mathbb{R}^{M_k \times 1}$ is the target vector to be estimated by agent k . The variable $\mathbf{v}_k(i) \in \mathbb{R}$ is a zero-mean white-noise process with variance $\mathbb{E} \mathbf{v}_k^2(i) = \sigma_{v,k}^2$ and assumed to be spatially independent. We further assume that the random processes $\mathbf{u}_{k,i}$ and $\mathbf{v}_\ell(j)$ are spatially and temporally independent for any k, ℓ, i , and j . We assume that each entry of w_k^o is determined by a grand target vector $w^o \in \mathbb{R}^{M \times 1}$, i.e., the relation between w_k^o and w^o can be described by

$$w_k^o = D_k w^o \quad (2)$$

where D_k is a matrix of size $M_k \times M$ and defined as

$$D_k(s, m) = \begin{cases} 1, & \text{if } w_k^o(s) \leftarrow w^o(m) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

The notation $x \leftarrow y$ denotes that the value of y is assigned to x . We are therefore considering a distributed inference problem where each agent has partial information about a grand target vector, i.e., the data at each agent is influenced by only some entries of w^o . Observe that the size M_k of the vector w_k^o is allowed to change with the node index, k , so that some agents may be influenced by more entries than other agents.

Now, given a network topology, two neighboring agents $\{k, \ell\}$ may share one or more common target entries, e.g., there can exist some index $m \in \{1, \dots, M\}$ such that

$$w_k^o(s) \leftarrow w^o(m), \quad w_\ell^o(s') \leftarrow w^o(m), \quad \ell \in \mathcal{N}_k \setminus \{k\} \quad (4)$$

where \mathcal{N}_k is the neighborhood of agent k . We are therefore motivated to consider the following constrained optimization problem for each agent k :

$$\begin{aligned} \min_{w_k} \quad & J_k(w_k) \triangleq \frac{1}{2} \mathbb{E} |\mathbf{d}_k(i) - \mathbf{u}_{k,i} w_k|^2 \\ \text{subject to} \quad & w_k(s) = w_\ell(s'), \quad \ell \in \mathcal{N}_k \setminus \{k\}, \\ & s \in \{1, \dots, M_k\}, \quad s' \in \{1, \dots, M_\ell\} \end{aligned} \quad (5)$$

The indices s and s' in (5) refer only to the common entries in w_k^o and w_ℓ^o . We provide an example in Fig. 1 to illustrate the setting defined in (5). For example, agents #1 and #2 share the common target entry $w^o(2)$; it is the leading element in the target vector for agent #2 and the trailing element in the target vector for agent #1. Observe that agents can share target entries even if they are not neighbors, as is the case with agents #1 and #3; they both share entry $w^o(1)$. However, the constraints for the common target entries can only exist between neighboring agents. For convenience and for later use, we collect the constraints for each agent k into a constraint set \mathbb{S}_k :

$$\mathbb{S}_k \triangleq \left\{ (\ell, s, s') \left| \begin{array}{l} w_k(s) = w_\ell(s'), \ell \in \mathcal{N}_k \setminus \{k\} \\ s \in \{1, \dots, M_k\}, \quad s' \in \{1, \dots, M_\ell\} \end{array} \right. \right\} \quad (6)$$

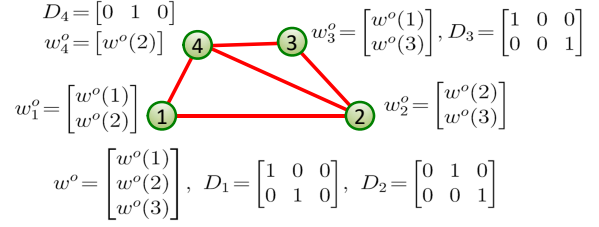


Fig. 1. An example to illustrate distributed inference under partial information exchange.

3. PENALTY-BASED LEARNING

One way to solve problem (5) is to reformulate it using penalty functions. Specifically, instead of solving (5), we consider the penalized version:

$$\min_{w_k} \quad J_k^p(w_k) \triangleq \frac{1}{2} \mathbb{E} |\mathbf{d}_k(i) - \mathbf{u}_{k,i} w_k|^2 + p_k(w_k) \quad (7)$$

where the notation $w^k \triangleq \text{col}\{w_\ell; \ell \in \mathcal{N}_k\}$ aggregates all unknowns in the neighborhood \mathcal{N}_k and $p_k(w^k)$ is a quadratic penalty function used to penalize agent k when any constraint $w_k(s) = w_\ell(s')$ is violated, i.e.,

$$p_k(w^k) \triangleq \sum_{(\ell, s, s') \in \mathbb{S}_k} \rho_k(\ell, s, s') \cdot [w_k(s) - w_\ell(s')]^2 \quad (8)$$

where $\rho_k(\ell, s, s')$ is a positive penalty parameter used to control the punishment level of violating the constraint $w_k(s) = w_\ell(s')$. Other choices for the penalty function are possible. It is sufficient for our purposes in this article to illustrate the main construction and results using (8).

3.1. Entry-Wise Diffusion Implementation

Following the approach from [18], the optimization problem (7) can be solved in two incremental steps: we first adapt with respect to $J_k(w_k)$ and then adapt with respect to $p_k(w^k)$. For this purpose, we start by noting that the gradient vector of $J_k(w_k)$ with respect to w_k is given by

$$\nabla_{w_k}^\top J_k(w_k) = R_{u,k} w_k - r_{du,k} \quad (9)$$

where $r_{du,k} = \mathbb{E} \mathbf{d}_k(i) \mathbf{u}_{k,i}^\top$. When the gradient of $J_k(w_k)$ is unavailable, we can approximate it by using the instantaneous approximations $r_{du,k} \approx \mathbf{d}_k(i) \mathbf{u}_{k,i}^\top$ and $R_{u,k} \approx \mathbf{u}_{k,i}^\top \mathbf{u}_{k,i}$:

$$\widehat{\nabla}_{w_k}^\top J_k(w_k) = \mathbf{u}_{k,i}^\top [\mathbf{u}_{k,i} w_k - \mathbf{d}_k(i)] \quad (10)$$

By doing so, we arrive at the adapt-then-penalize (ATP) diffusion strategy:

$$\text{(ATP)} \quad \begin{cases} \psi_{k,i} = w_{k,i-1} + \mu_k \mathbf{u}_{k,i}^\top [\mathbf{d}_k(i) - \mathbf{u}_{k,i} w_{k,i-1}] & (11) \\ w_{k,i} = \psi_{k,i} - \mu_k \nabla_{w_k}^\top p_k(\psi_{k,i}^k) & (12) \end{cases}$$

where $\psi_i^k \triangleq \text{col}\{\psi_{\ell,i}; \ell \in \mathcal{N}_k\}$. By differentiating the penalty function $p_k(\psi_i^k)$ with respect to w_k , it can be verified

that step (12) can be rewritten as

$$\mathbf{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} A_{\ell k}^\top \boldsymbol{\psi}_{\ell,i} \quad (13)$$

where $A_{\ell k}$ is the $M_\ell \times M_k$ matrix with entries defined by

$$A_{\ell k}(s', s) = \begin{cases} 2\mu_k \rho_k(\ell, s, s'), & \text{if } (\ell, s, s') \in \mathbb{S}_k \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

for $k \neq \ell$, and

$$A_{kk} = \text{diag} \left\{ 1 - \sum_{(\ell, 1, s') \in \mathbb{S}_k} 2\mu_k \rho_k(\ell, 1, s'), \dots, 1 - \sum_{(\ell, M_k, s') \in \mathbb{S}_k} 2\mu_k \rho_k(\ell, M_k, s') \right\} \quad (15)$$

Note that

$$[A_{1k}^\top \dots A_{Nk}^\top] \mathbf{1} = \mathbf{1} \quad (16)$$

where $\mathbf{1}$ denotes the vector with all one entries. Observe that the matrix $A_{\ell k}$ defines the combination weights between agent k and the common entries with agent ℓ from its neighborhood; we assume the step-sizes $\{\mu_k\}$ are sufficiently small such that the diagonal entries in $\{A_{kk}\}$ are nonnegative so that $A = [A_{\ell k}]$ is a left-stochastic matrix. It turns out that the particular forms (14) and (15) are not critical. It is sufficient to select an arbitrary left-stochastic matrix A as long as the zero-structure of its block components $\{A_{\ell k}\}$ and property (16) are satisfied. We can also switch the order of the incremental steps in (11)–(12) and arrive at the penalize-then-adapt (PTA) diffusion strategy:

$$\text{(PTA)} \begin{cases} \boldsymbol{\psi}_{k,i} = \sum_{\ell \in \mathcal{N}_k} A_{\ell k}^\top \mathbf{w}_{\ell,i-1} \end{cases} \quad (17)$$

$$\mathbf{w}_{k,i} = \boldsymbol{\psi}_{k,i} + \mu_k \mathbf{u}_{k,i}^\top [\mathbf{d}_k(i) - \mathbf{u}_{k,i} \boldsymbol{\psi}_{k,i}] \quad (18)$$

We remark that in the penalty steps (13) and (17), agents are only required to exchange a subset of the entries of their iterates with their neighboring agents, namely, those entries that define the constraints. This property reduces communication overhead, compared with traditional diffusion ATC and CTA algorithms in [10]. The ATP and PTA formulations (11)–(13) and (17)–(18) define a useful class of distributed strategies that include other important cases as special instances. For example, if for each agent k we set $D_k = I_M$ in (2) and extend the constraint set \mathbb{S}_k to cover all possible entries s and s' , we will arrive at the group diffusion LMS algorithm proposed in [19] for agents to assign different combination weights to different entries. If we further set $A_{\ell k} = a_{\ell k} I$, we will get the traditional diffusion LMS algorithms from [10].

4. PERFORMANCE ANALYSIS

We consider the ATP diffusion strategy (11)–(13). We can rewrite the adaptation step in (11) as

$$\boldsymbol{\psi}_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \mathbf{u}_{k,i}^\top \mathbf{u}_{k,i} (w_k^o - \mathbf{w}_{k,i-1}) + \mu_k \mathbf{u}_{k,i}^\top \mathbf{v}_k(i) \quad (19)$$

where we used the linear model (1). Collecting the iterates $\{\boldsymbol{\psi}_{k,i}\}$ and $\{\mathbf{w}_{k,i}\}$ from the across the network at time i into the aggregate vectors $\boldsymbol{\psi}_i = \text{col}\{\boldsymbol{\psi}_{1,i}, \dots, \boldsymbol{\psi}_{N,i}\}$ and $\mathbf{w}_i = \text{col}\{\mathbf{w}_{1,i}, \dots, \mathbf{w}_{N,i}\}$, we find that these network vectors evolve according to the dynamics:

$$\boldsymbol{\psi}_i = \mathbf{w}_{i-1} + \mathcal{M} \mathcal{R}_i (w_\star^o - \mathbf{w}_{i-1}) + \mathcal{M} \mathbf{s}_i \quad (20)$$

$$\mathbf{w}_i = \mathcal{A}^\top \boldsymbol{\psi}_i \quad (21)$$

where

$$\mathcal{M} \triangleq \text{diag}\{\mu_1 I_{M_1}, \dots, \mu_N I_{M_N}\} \quad (22)$$

$$\mathcal{R}_i \triangleq \text{diag}\{\mathbf{u}_{1,i}^\top \mathbf{u}_{1,i}, \dots, \mathbf{u}_{N,i}^\top \mathbf{u}_{N,i}\} \quad (23)$$

$$\mathbf{w}_\star^o \triangleq \text{col}\{w_1^o, \dots, w_N^o\} = \mathcal{D} w^o \quad (24)$$

$$\mathcal{D} \triangleq \begin{bmatrix} D_1 \\ \vdots \\ D_N \end{bmatrix}, \quad \mathbf{s}_i \triangleq \begin{bmatrix} \mathbf{u}_{1,i}^\top \mathbf{v}_1(i) \\ \vdots \\ \mathbf{u}_{N,i}^\top \mathbf{v}_N(i) \end{bmatrix}, \quad \mathcal{A} \triangleq \begin{bmatrix} A_{11} & \dots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \dots & A_{NN} \end{bmatrix} \quad (25)$$

Similarly, the network recursion for PTA diffusion is given by

$$\boldsymbol{\psi}_i = \mathcal{A}^\top \mathbf{w}_{i-1} \quad (26)$$

$$\mathbf{w}_i = \boldsymbol{\psi}_i + \mathcal{M} \mathcal{R}_i (w_\star^o - \boldsymbol{\psi}_i) + \mathcal{M} \mathbf{s}_i \quad (27)$$

We can capture the dynamics of both algorithms in a single unified model by introducing intermediate iterates $\phi_{k,i}$ and matrices $\{\mathcal{A}_1, \mathcal{A}_2\}$ and then writing:

$$\phi_i = \mathcal{A}_1^\top \mathbf{w}_{i-1} \quad (28)$$

$$\boldsymbol{\psi}_i = \phi_i + \mathcal{M} \mathcal{R}_i (w_\star^o - \phi_i) + \mathcal{M} \mathbf{s}_i \quad (29)$$

$$\mathbf{w}_i = \mathcal{A}_2^\top \boldsymbol{\psi}_i \quad (30)$$

The PTA case corresponds to the choice $\mathcal{A}_1 = \mathcal{A}$ and $\mathcal{A}_2 = I$ while the ATP case corresponds to $\mathcal{A}_1 = I$ and $\mathcal{A}_2 = \mathcal{A}$. To continue, we note the following useful properties:

$$\mathcal{A}_2 \mathcal{A}_1 = \mathcal{A}, \quad \mathcal{A}^\top \mathcal{D} = \mathcal{D}, \quad \mathcal{A}_1^\top \mathcal{D} = \mathcal{D}, \quad \mathcal{A}_2^\top \mathcal{D} = \mathcal{D}, \quad \mathcal{A}^\top \mathbf{1} = \mathbf{1} \quad (31)$$

From (24) and (28) we can write

$$w_\star^o - \phi_i = \mathcal{D} w^o - \mathcal{A}_1^\top \mathbf{w}_{i-1} = \mathcal{A}_1^\top (w_\star^o - \mathbf{w}_{i-1}) \quad (32)$$

and, similarly,

$$w_\star^o - \mathbf{w}_i = \mathcal{A}_2^\top (w_\star^o - \boldsymbol{\psi}_i) \quad (33)$$

Therefore, we arrive at the network error recursions:

$$\tilde{\phi}_i = \mathcal{A}_1^\top \tilde{\mathbf{w}}_{i-1} \quad (34)$$

$$\tilde{\boldsymbol{\psi}}_i = \tilde{\phi}_i - \mathcal{M} \mathcal{R}_i \tilde{\phi}_i - \mathcal{M} \mathbf{s}_i \quad (35)$$

$$\tilde{\mathbf{w}}_i = \mathcal{A}_2^\top \tilde{\boldsymbol{\psi}}_i \quad (36)$$

where $\tilde{\phi}_i \triangleq w_\star^o - \phi_i$ and similarly for the other error vectors. Combining (34)–(36) we conclude that the error vector $\tilde{\mathbf{w}}_i$ evolves according to the recursion:

$$\tilde{\mathbf{w}}_i = \mathcal{B}_i \tilde{\mathbf{w}}_{i-1} - \mathcal{G}_i \mathbf{s}_i \quad (37)$$

where $\mathcal{B}_i \triangleq \mathcal{A}_2^\top (I - \mathcal{M} \mathcal{R}_i) \mathcal{A}_1^\top$ and $\mathcal{G} \triangleq \mathcal{A}_2^\top \mathcal{M}$. It was shown in [11] that the traditional ATC and CTA diffusion algorithms have a network error recursion of the same general form as (37), except that now we have one critical difference. Expression (37) is more general and allows agents to have *different* sizes for their target vectors $\{w_k^o\}$. Furthermore, the matrices \mathcal{A}_1 and \mathcal{A}_2 now reflect refined connections: two connected agents only share a subset of their entries, which can be a single entry in the extreme case. Therefore, cooperation between agents is limited to entry-wise exchanges, as opposed to full vector exchanges in traditional implementations. Following similar arguments to those in [11], we can derive conditions on the step-size parameters to ensure mean-square convergence and stability. Proofs are omitted for brevity. For any nonnegative symmetric matrix Σ , we let $\sigma = \text{vec}(\Sigma)$ and use the compact notation $\|x\|_\sigma^2$ to refer to the squared weighted quantity $x^\top \Sigma x$.

Theorem 1. (Mean-square-error stability) *For sufficiently small step-sizes, i.e., for $\mu_k < \mu_o$ for some small enough μ_o , it holds that $\mathbb{E} \tilde{w}_i \rightarrow 0$, such that the estimates are asymptotically unbiased. Moreover, the weighted error variance satisfies the recursion:*

$$\mathbb{E} \|\tilde{w}_i\|_\sigma^2 = \mathbb{E} \|\tilde{w}_{i-1}\|_{\mathcal{F}\sigma}^2 + [\text{vec}(\mathcal{Y}^\top)]^\top \sigma \quad (38)$$

where $\mathcal{F} \triangleq \mathbb{E}(\mathcal{B}_i^\top \otimes \mathcal{B}_i^\top)$ with the Kronecker product \otimes , $\mathcal{Y} \triangleq \mathcal{G} \mathcal{S} \mathcal{G}^\top$, and

$$\mathcal{S} \triangleq \text{diag}\{\sigma_{v,1}^2 R_{u,1}, \dots, \sigma_{v,1}^2 R_{u,N}\} \quad (39)$$

□

5. SIMULATIONS

We consider a network with $N = 10$ agents. Each agent k is estimating a target vector w_k^o , which is a subvector of the grand target vector w^o of size $M = 10$. For each agent k , we assume that $R_{u,k}$ is diagonal and its diagonal entries are determined by a grand diagonal covariance matrix R_u , i.e.,

$$R_{u,k} = D_k R_u D_k^\top \quad (40)$$

Figure 2 shows the entries of the grand target vector w^o , the diagonal entries of the grand covariance matrix R_u , and the noise variance $\{\sigma_{v,k}^2\}$ at the agents. The network topology and the relations between $\{w_k^o\}$ and w^o are shown in Fig. 3. We set the step-size to $\mu_k = \mu = 0.005$ and the penalty parameter to $\rho_k(\ell, s, s') = \rho = 30$.

In Fig. 4, we simulate the learning curves of instantaneous network mean-square deviation (MSD), which is defined as

$$\text{MSD}_i \triangleq \frac{1}{N} \sum_{k=1}^N \mathbb{E} \|w_k^o - w_{k,i}\|^2 \quad (41)$$

We observe that both diffusion ATP and PTA algorithms exhibit better steady-state MSD performance than the non-cooperative case without imposing constraints and penalties. To examine the individual performance, we compare the steady-state individual MSD for each agent in Fig. 5. It is seen that all agents benefit from exchange of information with neighbors. The difference between diffusion ATP and PTA algorithms is negligible in the figures.

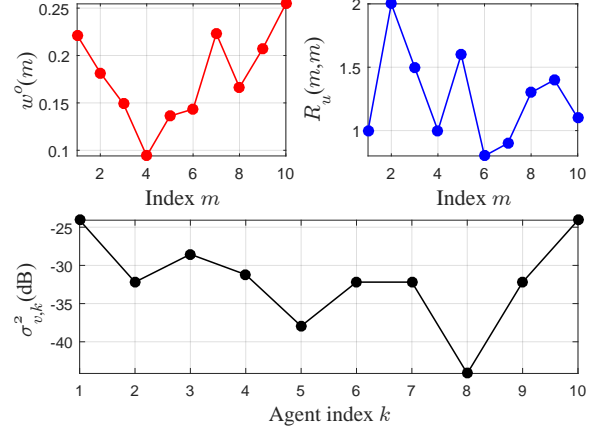


Fig. 2. Entries of w^o , R_u , and noise variance $\{\sigma_{v,k}^2\}$ used in the simulations.

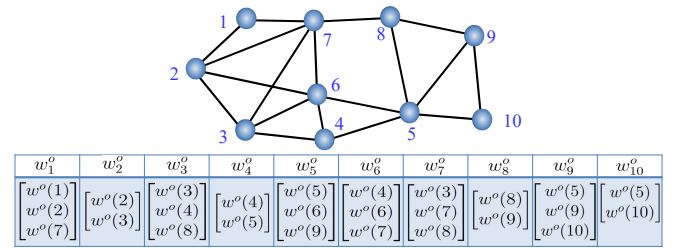


Fig. 3. Network topology and target vectors $\{w_k^o\}$.

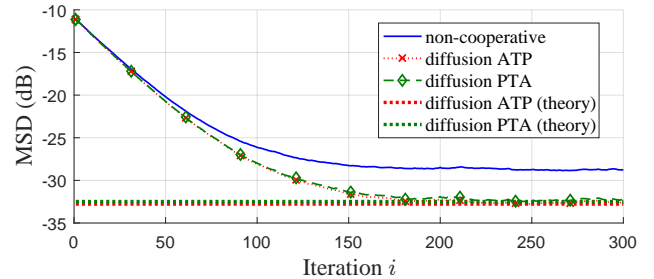


Fig. 4. Learning curves for the network mean-square-error deviation (MSD).

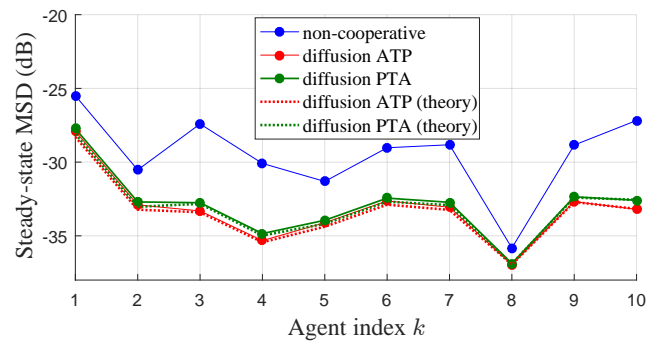


Fig. 5. Steady-state MSD for individual agents.

6. REFERENCES

- [1] P. Braca, S. Marano, and V. Matta, "Enforcing consensus while monitoring the environment in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3375–3380, July 2008.
- [2] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.
- [3] S. Sardellitti, M. Giona, and S. Barbarossa, "Fast distributed average consensus algorithms based on advection-diffusion processes," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 826–842, Feb. 2010.
- [4] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
- [5] S. Kar and J. M. F. Moura, "Convergence rate analysis of distributed gossip (linear parameter) estimation: Fundamental limits and tradeoffs," *IEEE J. Sel. Topics in Signal Process.*, vol. 5, no. 5, pp. 674–690, Aug. 2011.
- [6] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
- [7] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, no. 1, pp. 65–78, Sept. 2004.
- [8] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, June 2006.
- [9] A. H. Sayed, "Adaptive networks," *Proceedings of the IEEE*, vol. 102, no. 4, pp. 460–497, Apr. 2014.
- [10] A. H. Sayed, *Adaptation, Learning, and Optimization over Networks*, Foundations and Trends in Machine Learning, vol. 7, issue 4-5, NOW Publishers, 2014.
- [11] A. H. Sayed, "Diffusion adaptation over networks," in *Academic Press Library in Signal Processing*, R. Chellapa and S. Theodoridis, Eds., vol. 4, pp. 323–454, Academic Press, Elsevier, 2014.
- [12] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4692–4707, Oct. 2011.
- [13] J. Chen and A. H. Sayed, "On the learning behavior of adaptive networks — Part I: Transient analysis," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3487–3517, June 2015.
- [14] J. Chen and A. H. Sayed, "On the learning behavior of adaptive networks — Part II: Performance analysis," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3518–3548, June 2015.
- [15] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4129–4144, Aug. 2014.
- [16] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Multitask diffusion adaptation over asynchronous networks," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2835–2850, June 2016.
- [17] R. Arablouei, S. Werner, Y. F. Huang, and K. Doanay, "Distributed least mean-square estimation with partial diffusion," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 472–484, Jan. 2014.
- [18] Z. J. Towfic and A. H. Sayed, "Adaptive penalty-based distributed stochastic convex optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3924–3938, Aug. 2014.
- [19] C. Richard J. Chen, S. K. Ting and A. H. Sayed, "Group diffusion LMS," in *Proc. IEEE ICASSP*, Shanghai, China, Mar. 2016, pp. 4925–4929.