Distributed Optimization Over Markovian Switching Random Network

Peng Yi\textsuperscript{a,b}, Li Li\textsuperscript{a,b}

Abstract—In this paper, we investigate the distributed convex optimization problem over a multi-agent system with Markovian switching communication networks. The objective function is the sum of each agent’s local objective function, which cannot be known by other agents. The communication network is assumed to switch over a set of weight-balanced directed graphs with a Markovian property. We propose a consensus sub-gradient algorithm with two time-scale step-sizes to handle the uncertainty due to the Markovian switching topologies and the absence of global gradient information. With a proper selection of step-sizes, we prove the almost sure convergence of all agents’ local estimates to the same optimal solution when the union graph of the Markovian network’s states is strongly connected and the Markovian network is irreducible. Simulations are given for illustration of the results.

I. INTRODUCTION

There is an increasing research interest in distributed optimization over multiagent systems due to its broad applications in engineering networks, such as distributed parameters estimation in sensor networks \cite{1}, resource allocation in communication networks, \cite{2}, \cite{3}, and optimal power flow in power grids, \cite{4}, \cite{5}. Due to the privacy of each agent’s local data and the burden of data centralization, in distributed optimization problems each agent can only manipulate its local objective function without knowing other agents’ objective functions, while the global objective function to be optimized is usually taken as the sum of agents’ local objective functions. Many significant distributed optimization algorithms have been proposed and analyzed, including (sub)gradient algorithms \cite{6}, \cite{7}, dual averaging algorithms \cite{8}, primal-dual methods \cite{9}, \cite{10}, quantized algorithms \cite{11}, continuous-time algorithms \cite{12}, gradient tracking methods \cite{13}, \cite{14}. Please refer to \cite{15}–\cite{20} for the survey of recent developments in distributed optimization.

In distributed optimization, the agents must cooperatively find a consensual optimal solution by sharing information locally with network neighbors, hence, communication plays a vital role in the design and analysis of distributed optimization algorithm. Different communication models and graph connectivity assumptions, either deterministic or stochastic, have been discussed for different algorithms including uniformly joint strongly connected graphs \cite{6}, \cite{9}, quantized communication \cite{7}, random graphs \cite{9}, \cite{10}, broadcasting \cite{11} and gossip communication \cite{12}. In fact, the practical communication networks are essentially random and stochastic due to link failure, uncertain quantization, packet dropout or node recreation. Random communication networks with temporal independence assumptions have been investigated in distributed optimization. \cite{13} established the almost sure convergence of the consensus subgradient algorithm to an optimal point when the agents share information through independent broadcast communications. \cite{14} provided the almost sure convergence results for distributed subgradient algorithm when the communication link failures are independent and identically distributed over time. \cite{15} investigated the asynchronous distributed gradient method with a linear convergence rate for strongly convex functions when the graph weights are independently and identically drawn from the same probability space. \cite{16} proved the optimal convergence rate of distributed stochastic gradient methods for strongly convex functions over temporally independent identically distributed random networks. \cite{17} investigated the asymptotic normality and efficiency of distributed primal-dual gradient algorithm for independent and identically random communication networks. \cite{18} gave a primal-dual algorithm for distributed resource allocation, also with independent and identically random communication networks.

Nevertheless, the practical communications over multiagent systems are usually random but with temporal correlation. Markovian switching graphs have been adopted for modelling the random communication with one-step temporal dependence. For example, \cite{19}–\cite{21} have investigated the performance of averaging consensus algorithm with Markovian switching communication networks, \cite{22} have considered the distributed parameter estimation problem over Markovian switching topologies, and \cite{23} investigated the Kalman filter with Markovian packet losses when transmitting the measurements to the filter. However, to the best of our knowledge, how to achieve distributed optimization with Markovian switching graphs is not fully investigated, because distributed optimization is a fundamentally different task from consensus or parameter estimations, except that \cite{24} have studied distributed optimization over a switching state-dependent graphs. We also note that \cite{25}...
investigated the distributed optimization through the fixed points iteration of random operators derived from a general class of random graphs.

Motivated by the above, we investigate the consensus subgradient algorithm to achieve optimal consensus with Markovian switching topologies. The communication graph among the agents switches within a finite graph set following a Markovian chain. Note that [32] assumed that the random link failure is dependent on the node state rather than the previous step communication, hence, it considered a different Markovian model from the Markovian random graph considered here. We propose to select two different step-sizes for the consensus term and the gradient term to balance the speed of consensus and innovation. We find a sufficient choice of step-sizes to ensure that the consensus term is slightly “faster” than the innovation gradient term, and then we can give a mean consensus error bounds under the Markovian assumption. With these error bounds, we prove that all the agents converge to the same optimal solution with probability 1.

The paper is organized as follows. We give the formulation of the distributed optimization problem and Markovian switching communication model in Section II. We give the algorithm and sketch the main results in Section III. We give an illustrative numerical example in Section IV, and present the conclusions in section V.

**Notations:** Denote \(1_m = (1, ..., 1)^T \in \mathbb{R}^m\) and \(0_m = (0, ..., 0)^T \in \mathbb{R}^m\). For a column vector \(x \in \mathbb{R}^m\), \(x^T\) denotes its transpose. \(J_n\) denotes the identity matrix in \(\mathbb{R}^{n \times n}\). For a matrix \(A = [a_{ij}] \in \mathbb{R}^{N \times N}\), \(a_{ij}\) stands for the \((i, j)_{th}\) entry in \(A\). A matrix \(A\) is nonnegative if \(a_{ij} \geq 0\)\(, \forall i, j = 1, \cdots, N\). A nonnegative matrix \(A\) is called row stochastic iff \(A 1_N = 1_N\), and column stochastic matrix iff \(1_N^T A = 1_N^T\), while \(A\) is doubly stochastic iff \(A\) is both row and column stochastic. \(\otimes\) stands for the Kronecker product of two matrices. For a probability space \((\Xi, F, \mathbb{P})\), \(\Xi\) is the sample space, \(F\) is the \(\sigma\)-algebra and \(\mathbb{P}\) is the probability measure. For \(k = 0, 1, 2, \cdots, (v_k, F_k)\) is an adapted sequences if \(\sigma(v_k) \in F_k\) for all \(k\). The expectation of a random variable is denoted as \(\mathbb{E}[\cdot]\).

A directed graph \(G = \{V, \mathcal{E}_G, A_G\}\) is defined with node set \(V = \{1, ..., N\}, \) edge set \(\mathcal{E}_G \subset \mathcal{V} \times \mathcal{V}\), and adjacency matrix \(A_G = [a_{ij}] \in \mathbb{R}^{N \times N}\). \(a_{ij}\) stands for the \((i, j)_{th}\) entry of \(A_G\) if and only if \(i\) can get information from \(j\). \(A_G = [a_{ij}]\) is nonnegative and row stochastic, and \(0 < a_{ij} \leq 1\) if \((j, i) \in \mathcal{E}_G\), and \(a_{ij} = 0\), otherwise. Denote by \(\mathcal{N}_i = \{j | (j, i) \in \mathcal{E}_G\}\) the neighbor set of agent \(i\). A path of graph \(G\) is a sequence of distinct agents in \(V\) such that any consecutive agents in the sequence corresponding to an edge of the graph \(G\). Agent \(j\) is said to be connected to agent \(i\) if there is a path from \(j\) to \(i\). Graph \(G\) is strongly connected if any two agents are connected. Graph \(G\) is called weighted-balanced if adjacency matrix \(A\) is doubly stochastic, i.e., \(1_N^T A_G = 1_N^T A_G^T\). Denote by \(D_G = \text{diag}\{\sum_{j=1}^N a_{ij}, ..., \sum_{j=1}^N a_{Nj}\}\), called the in-degree matrix of \(G\). Then, the (weighted) Laplacian matrix of \(G\) is \(L_G := D_G - A_G\). When graph \(G\) is strongly connected, 0 is a simple eigenvalue of Laplacian \(L_G\) with the eigenspace \(\{\alpha 1_N | \alpha \in \mathbb{R}\}\).

**II. PROBLEM FORMULATION**

In this section, we formulate the distributed optimization problem.

Consider a multi-agent network with agent (node) set \(V = \{1, ..., N\}\), where agent \(i\) has its own objective function \(f_i(x)\) unknown to any other agents. The task is to find the optimal solution of the sum of all the local objective functions, that is,

\[
\min_{x \in \mathbb{R}^n} f(x), \quad f(x) = \sum_{i=1}^N f_i(x),
\]

where \(f_i : \mathbb{R}^n \rightarrow \mathbb{R}\), as a lower semicontinuous (possible nonsmooth) convex function, is the local objective function of agent \(i\), and \(f(.)\) is the global objective function. We give the following assumption on the objective functions.

**Assumption 1:** 1) The optimization problem in (1) is solvable, i.e., there exists a finite \(x^* \in \mathbb{R}^n\) such that

\[
x^* \in X^* \triangleq \arg \min f(x), \quad f(x) = \sum_{i=1}^N f_i(x).
\]

2) The sub-gradient sets of \(f_i(x)\), are uniformly bounded for all \(i \in V\), i.e., there exists a constant \(l\) such that

\[
\forall g(x) \in \partial f_i(x), \quad \|g(x)\| \leq l, \forall x \in \text{dom}(f_i), \forall i \in V.
\]

We assume the agents exchange information locally through a Markovian switching random communication network. All the possible communication topologies form a set of a finite number of graphs: \(\{G_1, \cdots, G_m\}\) with each graph endowed with an adjacency matrix \(A_G\). The time is slotted as \(k = 1, 2, \cdots\). And then, we use a random process \(\theta(k)\), which is a Markovian chain on a finite index set \(\mathcal{I} = \{1, ..., m\}\) with a stationary transition matrix \(P = [p_{ij}] \in \mathbb{R}^{m \times m}\), to indicate the communication graph at time \(k\), i.e., \(G(k) = G_i\) when \(\theta(k) = i\). The markovian property of \(\theta(k)\) implies that given the graph at time \(k\) being \(G_i\), the probability of the communication graph at time \(k+1\) being \(G_j\) is \(p_{ij}\). The works about average consensus in [27], [28], [30] have provided detailed descriptions and motivations for using Markovian switching communication networks in distributed computation over multi-agent systems, including wireless sensor networks and UAV swarms.

Here is the assumption on the Markovian communication graphs:

**Assumption 2:** 1) The adjacency matrices \(A_G\) of each graph in the set \(\{G_1, \cdots, G_m\}\) is a doubly stochastic matrix, and the union graph

\[
G_c \triangleq \bigcup_{i=1}^m G_i = \{V, \bigcup_{i=1}^m \mathcal{E}_{G_i}, \frac{1}{m} \sum_{i=1}^m A_{G_i}\}
\]
is strongly connected.

2) The Markovian chain $\theta(k)$ is irreducible.

III. DISTRIBUTED ALGORITHM AND MAIN RESULTS

In this section, we provide the algorithm with the main results.

Denote by $x_i(k) \in \mathbb{R}^n$ the estimate of agent $i$ for the optimal solution $x^*$ at time $k$. The random variable $\theta(k)$ evolves as a markovian chain. The communication graph takes $G(k) \triangleq G(\theta(k)) = (\mathcal{V}, \mathcal{E}_{G(\theta(k))}, A_{G(\theta(k))})$ at time $k$. Agent $i$ can get the estimates of its neighboring agents $N_i(k) = \{j|(j, i) \in \mathcal{E}_{G(\theta(k))}\}$ with $G(\theta(k))$. And then, each agent updates its estimate with the following algorithm.

Algorithm 1: Consensus subgradient algorithm

\textbf{Initialize:} Agent $i \in \mathcal{V}$ picks an initial state $x_i(0) \in \mathbb{R}^n$.

\textbf{Iterate until convergence}

At time $k$, each agent $i \in \mathcal{V}$ gets its neighbour states $\{x_j(k)\}_{j \in N_i(k)}$ through the random graph $G(k)$, and updates its local state as follows

$$x_i(k + 1) = x_i(k) + \alpha_k \sum_{j=1}^{N} a_{ij}(k)(x_j(k) - x_i(k)) - \beta_k d_i(k), \quad (2)$$

where $\alpha_k > 0$ and $\beta_k > 0$ are the step-sizes, $a_{ij}(k)$ is $(i,j)_{th}$ entry of $A_{G(\theta(k))}$, and $d_i(k) \in \partial f_i(x_i(k))$ is a (sub)gradient of $f_i(x)$ at $x_i(k)$.

Algorithm 1 is an extension of the distributed subgradient algorithm in [6], [7] by adding an additional step-size. In equation (2), the first consensus term drives each agent’s state towards the averaging of all agents’ states, while the second term provides the innovative gradient information to search for the optimal solution $x^*$.

To guarantee the algorithm convergence even with a randomly switching network, we have two different step-sizes $\alpha_k$ and $\beta_k$ to control the speed of consensus and innovation. In fact, we require that “consensus” speed is a bit of faster than “innovation” term as specified by the following assumption.

Assumption 3: We take the step-sizes in (2) as

$$\alpha_k = \frac{a_1}{(k + 1)^{a_1}}, \quad \beta_k = \frac{a_2}{(k + 1)^{a_2}}, \quad (3)$$

where $a_1 > 0$, $a_2 > 0$, $0 < \delta_1 < \delta_2 \leq 1$, and $\delta_2 - \delta_1 \geq \frac{1}{2}$.

Now we are ready to present the main analysis results for Algorithm 1.

Theorem 1 (Almost sure consensus): Suppose Assumptions 1, 2 and 3 hold. Let $x_i(k), i \in \mathcal{V}$ be generated by (2), and $y(k) = \frac{1}{N} \sum_{i=1}^{N} x_i(k)$. Then the following statements hold.

1) The agents’ states reach consensus and track the averaging of all the agents’ states asymptotically with probability 1, i.e.,

$$\lim_{k \to \infty} \| x_i(k) - y(k) \| = 0, \quad \forall i \in \mathcal{V}, \quad a.s. \quad (4)$$

2) The accumulation of the norm of track error $y(k) - x_i(k)$ weighted by the step-sizes $\beta_k$ is bounded for each agent, i.e.,

$$\sum_{k=1}^{\infty} \beta_k \| y(k) - x_i(k) \| < \infty, \quad \forall i \in \mathcal{V}, \quad a.s. \quad (5)$$

Remark 1: Theorem 1 shows that all the agents almost surely reach consensus asymptotically. In fact, we can also show the convergence rate for reaching consensus is dominated by the difference between $\delta_1, \delta_2$. Specifically, we can find a $\tau < \delta_2 - \frac{1}{2}$ such that

$$\lim_{k \to \infty} (k + 1)^{\tau} \| y(k) - x_i(k) \| = 0 \quad a.s., \forall i \in \mathcal{V}$$

Theorem 2 (Almost sure converge to a consensual solution): Suppose Assumptions 1.2 and 3 hold. Then with Algorithm 1, all the agents’ states almost surely converge to the same optimal solution of (1), i.e.,

$$\lim_{k \to \infty} x_i(k) = x^*, \forall i \in \mathcal{V}, \quad a.s.$$ Due to page limitation, the proofs can be found online in [34].

IV. SIMULATION

Example 1: We give an example to illustrate the algorithm. Consider five agents with the local objective functions as follows:

$$f_1(x) = \ln(e^{0.1x_1} + e^{0.2x_2}) + 5 \min_{z \in \Omega} \| x - z \|;$$

$$f_2(x) = 3(x_1)^2 \ln((x_1)^2 + 1) + 2(x_2)^2;$$

$$f_3(x) = 3(x_1 - 10)^2 + 0.2(x_2 - 8)^2 + 2|x_1| + 2|x_2|;$$

$$f_4(x) = \frac{4(x_1)^2}{\sqrt{2(x_1)^2 + 1}} + 0.1(x_1 + x_2)^2;$$

$$f_5(x) = (x_1 + 5x_2 - 10)^2;$$

$$+ 4\max\{x_1 + x_2, (x_1 + x_2)^2\},$$

with a decision variable as $x = (x_1, x_2) \in \mathbb{R}^2$ and a set $\Omega$ as $\Omega = \{x \in \mathbb{R}^2 | x_1^2 + x_2^2 \leq 1\}$.

The five agents share information with three graphs $\{G_1, G_2, G_3\}$, whose weighted adjacency matrices are $A_1, A_2, A_3 \in \mathbb{R}^{5 \times 5}$, respectively. The transition matrix of the stationary Markovian chain $\theta(k)$ is $P \in \mathbb{R}^{3 \times 3}$. We let $P, A_1, A_2, A_3$ to be the following matrices:

$$P = \begin{pmatrix} 0.5 & 0 & 0.5 \\ 0 & 0.6 & 0.4 \\ 0.2 & 0 & 0.8 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0.5 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 \end{pmatrix},$$

$$A_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0.5 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 \end{pmatrix}, \quad A_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0.5 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 \end{pmatrix}.$$
V. CONCLUSIONS

In this paper, we proposed a consensus subgradient algorithm to solve a distributed optimization problem with Markovian switching random communication networks. The algorithm was given with two time-scale stepsizes, different from most existing ones. We showed the almost sure convergence with a proper connectivity assumption and step-size choices. In the future, we will work on the mean-square convergence rate analysis.

REFERENCES

[1] J. Lei, H.-F. Chen, and H.-T. Fang, “Primal–dual algorithm for distributed constrained optimization,” Systems & Control Letters, vol. 96, pp. 110–117, 2016.
[2] E. Wei, A. Ozdaglar, and A. Jadbabaie, “A distributed newton method for network utility maximization–i: Algorithm,” IEEE Transactions on Automatic Control, vol. 58, no. 9, pp. 2162–2175, 2013.
[3] W. Shi, Q. Ling, G. Wu, and W. Yin, “Extra: An exact first-order algorithm for decentralized consensus optimization,” SIAM Journal on Optimization, vol. 25, no. 2, pp. 944–966, 2015.
[4] E. Dall’Anese, H. Zhu, and G. B. Giannakis, “Distributed optimal power flow for smart microgrids,” IEEE Transactions on Smart Grid, vol. 4, no. 3, pp. 1464–1475, 2013.
[5] P. Yi, Y. Hong, and F. Liu, “Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems,” Automatica, vol. 74, pp. 259–269, 2016.
[6] A. Nedic and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” IEEE Transactions on Automatic Control, vol. 54, no. 1, pp. 48–61, 2009.
[7] I. Lobel and A. Ozdaglar, “Distributed subgradient methods for convex optimization over random networks,” IEEE Transactions on Automatic Control, vol. 56, no. 6, pp. 1291–1306, 2010.
[8] J. C. Duchi, A. Agarwal, and M. J. Wainwright, “Dual averaging for distributed optimization: Convergence analysis and network scaling,” IEEE Transactions on Automatic Control, vol. 57, no. 3, pp. 592–606, 2011.
[9] J. Lei, H.-F. Chen, and H.-T. Fang, “Asymptotic properties of primal-dual algorithm for distributed stochastic optimization over imperfectly connected networks,” IEEE Journal on Selected Areas in Communications, vol. 36, no. 5, pp. 1169–1182, 2018.
[10] J. Lei, P. Yi, G. Shi, and B. D. Anderson, “Distributed algorithms with finite data rates that solve linear equations,” SIAM Journal on Optimization, vol. 30, no. 2, pp. 1191–1222, 2020.
[11] P. Yi, Y. Hong, and F. Liu, “Distributed gradient algorithm for constrained optimization with application to load sharing in power systems,” Systems & Control Letters, vol. 83, pp. 45–52, 2015.
[12] X. Zeng, P. Yi, and Y. Hong, “Distributed continuous-time algorithm for constrained convex optimizations via nonsmooth analysis approach,” IEEE Transactions on Automatic Control, vol. 62, no. 10, pp. 5227–5233, 2016.
[13] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes,” in 2015 54th IEEE Conference on Decision and Control (CDC), IEEE, 2015, pp. 2055–2060.
[14] S. Pu, W. Shi, J. Xu, and A. Nedić, “A push-pull gradient method for distributed optimization in networks,” in 2018 IEEE Conference on Decision and Control (CDC). IEEE, 2018, pp. 3385–3390.
[15] A. H. Sayed et al., “Adaptation, learning, and optimization over networks,” Foundations and Trends® in Machine Learning, vol. 7, no. 4-5, pp. 311–801, 2014.
[16] A. Nedić et al., “Convergence rate of distributed averaging dynamics and optimization in networks,” Foundations and Trends® in Systems and Control, vol. 2, no. 1, pp. 1–100, 2015.
[17] P. Yi and Y. Hong, “Distributed cooperative optimization and its applications,” Scientia Sinica Mathematica, vol. 46, no. 10, pp. 1547–1564, 2016.
[18] A. Nedić and J. Liu, “Distributed optimization for control,” Annual Review of Control, Robotics, and Autonomous Systems, vol. 1, pp. 77–103, 2018.
[19] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson, “A survey of distributed optimization,” Annual Reviews in Control, 2019.
[20] G. Notarstefano, I. Notarnicola, A. Camisa et al., “Distributed optimization for smart cyber-physical networks,” Foundations and Trends® in Systems and Control, vol. 7, no. 3, pp. 253–383, 2019.
[21] P. Yi and Y. Hong, “Quantized subgradient algorithm and data-rate analysis for distributed optimization,” IEEE Transactions on Control of Network Systems, vol. 1, no. 4, pp. 380–392, 2014.
[22] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Convergence of asynchronous distributed gradient methods over stochastic networks,” *IEEE Transactions on Automatic Control*, vol. 63, no. 2, pp. 434–448, 2017.
[23] A. Nedic, “Asynchronous broadcast-based convex optimization over a network,” *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1337–1351, 2010.
[24] J. Lu, C. Y. Tang, P. R. Regier, and T. D. Bow, “Gossip algorithms for convex consensus optimization over networks,” *IEEE Transactions on Automatic Control*, vol. 56, no. 12, pp. 2917–2923, 2011.
[25] D. Jakovetic, D. Bajovic, A. K. Sahu, and S. Kar, “Convergence rates for distributed stochastic optimization over random networks,” in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 4238–4245.
[26] P. Yi, J. Lei, and Y. Hong, “Distributed resource allocation over random networks based on stochastic approximation,” *Systems & Control Letters*, vol. 114, pp. 44–51, 2018.
[27] M. Huang, S. Dey, G. N. Nair, and J. H. Manton, “Stochastic consensus over noisy networks with markovian and arbitrary switches,” *Automatica*, vol. 46, no. 10, pp. 1571–1583, 2010.
[28] I. Matei, J. S. Baras, and C. Somarakis, “Convergence results for the linear consensus problem under markovian random graphs,” *SIAM Journal on Control and Optimization*, vol. 51, no. 2, pp. 1574–1591, 2013.
[29] T. Li and J. Wang, “Distributed averaging with random network graphs and noises,” *IEEE Transactions on Information Theory*, vol. 64, no. 11, pp. 7063–7080, 2018.
[30] Q. Zhang and J.-F. Zhang, “Distributed parameter estimation over unreliable networks with markovian switching topologies,” *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2545–2560, 2012.
[31] N. Xiao, L. Xie, and M. Fu, “Kalman filtering over unreliable communication networks with bounded markovian packet dropouts,” *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 19, no. 16, pp. 1770–1786, 2009.
[32] I. Lobel, A. Ozdaglar, and D. Feijer, “Distributed multi-agent optimization with state-dependent communication,” *Mathematical programming*, vol. 129, no. 2, pp. 255–284, 2011.
[33] S. S. Alaviani and N. Elia, “Distributed multi-agent convex optimization over random digraphs,” *IEEE Transactions on Automatic Control*, 2019.
[34] P. Yi and L. Li, “Distributed optimization over markovian switching random network,” *arXiv preprint arXiv:2002.09862*, 2020.