Learning-Augmented Decentralized Online Convex Optimization in Networks

PENGFEI LI, University of California, Riverside
JIANYI YANG, University of California, Riverside
ADAM WIERMAN, California Institute of Technology
SHAOLEI REN, University of California, Riverside

ACM Reference Format:
Pengfei Li, Jianyi Yang, Adam Wierman, and Shaolei Ren. 2023. Learning-Augmented Decentralized Online Convex Optimization in Networks. 1, 1 (May 2023), 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 PROBLEM FORMULATION

This paper studies the problem of decentralized online convex optimization in networks, where inter-connected agents must individually select actions with sequentially-revealed local online information and delayed feedback from their neighboring agents. We consider a setting where, at each step, agents must decide on an action using local information while collectively seeking to minimize a global cost consisting of the sum of (i) the agents' node costs, which capture the local instantaneous effects of the individual actions; (ii) temporal costs, which capture the (inertia) effects of local temporal action changes; and (iii) spatial costs, which characterize the loss due to unaligned actions of two connected neighboring agents in the network. This problem models a wide variety of networked systems with numerous applications, such as decentralized control in power systems, capacity allocation in multi-rack data centers, spectrum management in multi-user wireless networks, multi-product pricing in revenue management, among many others.

At step $t = 1, \cdots, T$, each agent v selects an irrevocable action $x_t^v \in \mathbb{R}^n$. We denote $x_t = [x_t^1, \cdots, x_t^V]$ as the action vector for all agents at step t, where the superscript v represents the agent index whenever applicable. After x_t is selected for step t, the network generates a global cost $g_t(x_t)$ which consists of the following three parts.

• **Node cost $f_t^v(x_t^v)$:** Each individual agent incurs a node cost $f_t^v(x_t^v)$, which only relies on the action of a single agent v at step t and measures the effect of the agent's decision on itself.

• **Temporal cost $c_t^v(x_t^v, x_{t-1}^v)$:** It couples the two temporal-adjacent actions of a single agent v and represents the effect of temporal interactions to smooth actions over time.

• **Spatial cost $s_t^{(v,u)}(x_t^v, x_t^u)$:** It is incurred if an edge exists between two agents v and u, capturing the loss due to unaligned actions of two connected agents.

Assumption 1.1. *The node cost $f_t^v\colon \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is convex and ℓ_f-smooth.*

Assumption 1.2. *The temporal interaction cost $c_t^v\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is convex and ℓ_T-smooth.*

Assumption 1.3. *The spatial interaction cost $s_t^{v,u}\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is convex and ℓ_S-smooth.*

The global costs over a sequence of T time steps defined as:

$$cost(x_{1:T}) = \sum_{t=1}^{T} g_t(x_t) = \sum_{t=1}^{T} \sum_{v \in \mathcal{V}} f_t^v(x_t^v) + \sum_{t=1}^{T} \sum_{v \in \mathcal{V}} c_t^v(x_t^v, x_{t-1}^v) + \sum_{t=1}^{T} \sum_{(v,u) \in \mathcal{E}} s_t^{(v,u)}(x_t^v, x_t^u).$$

Authors' addresses: Pengfei Li, University of California, Riverside; Jianyi Yang, University of California, Riverside; Adam Wierman, California Institute of Technology; Shaolei Ren, University of California, Riverside.

2023. XXXX-XXXX/2023/5-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

$$\sum_{\tau=1}^{t} f_\tau^v(x_\tau^v) + \sum_{\tau=1}^{t} c_\tau^v(x_\tau^v, x_{\tau-1}^v) + \sum_{\tau=1}^{t-1} \sum_{(v,u)\in\mathcal{E}} \kappa_\tau^{(v,u)} \cdot s_\tau^{(v,u)}(x_\tau^v, x_\tau^u) + R(x_t^v, x_t^{v,\dagger})$$
$$\le (1+\lambda)\Big(\sum_{\tau=1}^{t} f_\tau^v(x_\tau^{v,\dagger}) + \sum_{\tau=1}^{t} c_\tau^v(x_\tau^{v,\dagger}, x_{\tau-1}^{v,\dagger}) + \sum_{\tau=1}^{t-1} \sum_{(v,u)\in\mathcal{E}} \kappa_\tau^{(v,u)} \cdot s_\tau^{(v,u)}(x_\tau^{v,\dagger}, x_\tau^{u,\dagger})\Big) \quad (1)$$

2 MAIN RESULT

Our algorithm, LADO (Learning-Augmented Decentralized Online optimization), exploits ML predictions to improve average-case cost performance while still providing the adversarial robustness guarantees with respect to any given baseline/expert policy. The key idea behind LADO is to leverage a baseline policy to safeguard online actions to avoid too greedily following ML predictions that may not be robust. The difficulty in design is determining how to manage the spatial information inefficiency (i.e., not knowing the neighboring agents' actions in advance). To address this, we propose a novel spatial cost decomposition to split the shared spatial cost in an adaptive manner between connected agents so that each agent can safeguard its own actions based on local information. We also introduce temporal reservation costs to address the worst-case future uncertainties.

Specifically, we project the ML prediction $\tilde{x}_t^v$ such that it satisfies the convex set specified by Eqn. (1) for each agent v at time t. In the constraint, the weight $\kappa_t^{(v,u)}$ (attributed to agent v) for adaptively splitting the spatial cost $s_t^{(v,u)}$ between agent v and agent u is

$$\kappa_t^{(v,u)} = \frac{\|x_t^v - x_t^{v,\dagger}\|^2}{\|x_t^v - x_t^{v,\dagger}\|^2 + \|x_t^u - x_t^{u,\dagger}\|^2}, \quad (2)$$

and the reservation cost is

$$R(x_t^v, x_t^{v,\dagger}) = \frac{\ell_T + \ell_S \cdot D_v}{2}(1 + \frac{1}{\lambda_0})\|x_t^v - x_t^{v,\dagger}\|^2, \quad (3)$$

where ℓ_T and ℓ_S are smoothness parameters for the temporal and spatial cost functions, D_v is the degree of agent v (i.e., the number of agents connected to agent v), and $0 < \lambda_0 = \sqrt{1+\lambda} - 1$ is the optimal hyperparameter to adjust the size of the robust action set. Our main results provide worst-case and average cost bounds for LADO.

THEOREM 2.1. *(λ-robustness) Given any ML policy $\tilde{\pi}$ and expert policy $\pi^\dagger$, for any $\lambda > 0$, the cost of* LADO *satisfies $cost(\text{LADO}, g_{1:T}) \le (1+\lambda) \cdot cost(\pi^\dagger, g_{1:T})$ for any problem instance $g_{1:T} \in \mathcal{G}$.*

THEOREM 2.2 (AVERAGE COST OF LADO($\tilde{\pi}_\lambda^\circ$)). *Given the optimal projection-aware ML policy $\tilde{\pi}_\lambda^\circ$, for any $\lambda > 0$, by optimally setting $\lambda_0 = \sqrt{1+\lambda} - 1$, the average cost of* LADO($\tilde{\pi}_\lambda^\circ$) *is upper bounded by*

$$AVG(\text{LADO}(\tilde{\pi}_\lambda^\circ)) \le \min\left\{(1-\alpha_\lambda)AVG(\pi^\dagger) + \alpha_\lambda AVG(\tilde{\pi}^*), \left(\sqrt{AVG(\tilde{\pi}^*)} + \hat{\phi}_\lambda(\tilde{\pi}^*) \cdot \sqrt{AVG(\pi^\dagger)}\right)^2\right\} \quad (4)$$

where $AVG(\pi^\dagger)$ and $AVG(\tilde{\pi}^*)$ are the average costs of the expert policy and the optimal ML policy (without robustness constraint), respectively, the weight $\alpha_\lambda = \min\left\{(\sqrt{1+\lambda} - 1)\sqrt{\frac{2}{\ell_T + \ell_f + D\cdot\ell_S} \cdot \frac{1}{\hat{\rho}}}, 1\right\}$ with $\hat{\rho} = \max_{g_{1:T}, v\in\mathcal{V}, t\in[1,T]} \frac{\sum_{i=1}^t \|x_i^{v,\dagger} - \tilde{x}_i^{v,*}\|^2}{\sum_{i=1}^t f_i^v(x_i^{v,\dagger}) + c_i^v(x_i^{v,\dagger}, x_{i-1}^{v,\dagger})}$ being the multi-step action discrepancy between the expert policy and the optimal ML policy normalized by the expert policy's local cost, $\hat{\phi}_\lambda(\tilde{\pi}^*) = \left[\sqrt{\gamma\hat{\rho}} - \sqrt{1+\lambda} + 1\right]^+$ with $\gamma = \frac{2\ell_T + \ell_f + D\ell_S}{2}$, ℓ_T, ℓ_f, ℓ_S are the smoothness factors for the temporal, node and spatial costs, respectively, and $D = \max_v D_v$ is the maximum node degree in the network.