2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Decentralized Online Convex Programming with Local Information

Maxim Raginsky, Nooshin Kiarashi, and Rebecca Willett

Abstract— **This paper describes a novel approach to decentralized online optimization in a large network of agents. At each stage, the agents face a new objective function that reflects the effects of a changing environment, and each agent can share information pertaining to past decisions and cost functions only with his neighbors. These operating conditions arise in many practical applications, but introduce challenging questions related to the performance of distributed strategies relative to impractical centralized approaches. The proposed algorithm yields small regret (i.e., the difference between the total cost incurred using causally available information and the total cost that would have been incurred in hindsight had all the relevant information been available all at once) and is robust to evolving network topologies. It combines a subgradient-based sequential convex optimization scheme with decentralized decision-making via approximate dynamic programming.**

I. Introduction

Consider a network of agents who cooperate to accomplish a common objective. For instance, the agents may represent various providers in a power grid, and at each time must decide how to satisfy the temporally varying demands for power with minimal expense. Other examples would include sensor nodes in a wireless sensor network or various decision-makers in a large organization. In such settings, decentralized architectures, in which the agents do not rely on a central facility for relevant information but must instead communicate with other agents, are preferable to centralized ones for ensuring robustness against localized failures and interruptions, as well as for reducing overhead in data collection, transmission, storage, and processing.

One approach towards designing a strategy for the agents would be to formulate the problem at hand as a stochastic multistage decision process with a prescribed sequence of cost functions and a prescribed information structure [1], [2]. However, most complex networks are deployed in highly dynamic and uncertain environments that do not admit easily identifiable or tractable models. To handle this uncertainty robustly, we can instead represent the temporal evolution of the environment by means of an arbitrarily varying sequence of cost functions, where the cost function for each stage is revealed only *ex post facto*, after the relevant decisions had already been made. The goal then is to minimize *regret*, i.e., the difference between the total cost incurred using causally available information and the total cost that would have been incurred in hindsight had all the relevant information been available all at once. This *online optimization* approach, which has recently gained a lot of popularity in the machine learning community [3], has not yet been consistently applied to multiagent decision and control problems.[1]

In this paper, we consider the problem of decentralized online optimization in a large network from the viewpoint of sequential team decisions with nonclassical information structures. This problem has several salient features:

- *Time-varying objective functions* – the quantity to be optimized varies with time due to uncertain environment dynamics, and the agents must adapt to these variations.
- *Additive local costs* – the objective at each stage is a sum of local costs involving a small subset of agents.
- *Local information* – each agent has only partial and localized knowledge of past decisions of other agents, as well as of the past cost functions.

Our contribution is twofold: (1) We develop a general mathematical framework for decentralized online optimization, including the appropriate notion of regret, by extending the definition of *information structure* in the sense of Witsenhausen [1] to settings involving a sequence of arbitrarily varying cost functions, where each agent possesses partial knowledge not only of other agents' past actions and observations, but also of past cost functions. To the best of our knowledge, this is the first time cost functions have been considered as part of an information structure, although the need for such a framework had been pointed out before [5]. (2) We focus on a particular information structure, under which each agent receives information only from agents in a fixed-radius neighborhood. We give a constructive proof that, provided the neighborhood radius is sufficiently large as a function of the planning horizon (but *independent of the number of agents*), there exists an efficiently implementable strategy that achieves essentially the same regret as in the fully centralized case, and is furthermore robust to changes in the network topology. This information structure was first studied by Rusmevichientong and van Roy [6] in the context of single-stage decentralized optimization over finite decision spaces, and one of the byproducts of the present work is an extension of their results to multi-stage decentralized convex optimization problems over compact Euclidean domains.

A. Related work

The problem of decentralized optimization over a network has received considerable attention for some time, dating back to the seminal work of Tsitsiklis *et al.* [7], [8]. In this framework, the agents are located at the nodes of a graph,

This work was supported by NSF grant CCF-1017564 and by AFOSR grant FA9550-10-1-0390.

The authors are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA; e-mail: {m.raginsky, nooshin.kiarashi, willett}@duke.edu.

[1] However, it has been recently shown that many classical adaptive control techniques are instances of online convex optimization [4].

and each agent can communicate only with his neighbors. Each agent performs local averaging of his current decision with those of his neighbors, as dictated by a doubly stochastic matrix that conforms to the topology of the graph. Recent work by Nedić and Ozdaglar [9] and by Duchi *et al.* [10] applies this methodology to the problem of decentralized convex optimization in networks.

The above work assumes that the function to be minimized is fixed, and the agents communicate over multiple rounds. Closer to our own setting, Yan *et al.* [11] consider the problem of decentralized online optimization, in which the objective function changes arbitrarily between rounds. Their scheme is similar to [9] and [10] in its use of local averaging combined with a descent algorithm. However, methods that rely on graph-conformant stochastic matrices are only relatively robust against changes in the network topology (e.g., the algorithm of [10] is proven to be resilient against random and independent link failures, but it is not clear how it can handle nonergodic changes in the network structure). To handle a dynamic network topology, there must be a way to adjust the weight matrix in real time and in a decentralized manner, which may require considerable overhead.

By contrast, the approach developed in the present paper obviates the need for carefully designed stochastic matrices. Instead, it highlights the importance of *common randomness* in guaranteeing that, at least with certain types of information structures, the decisions can be made by individual agents independently of "faraway" agents, yet be almost as good as the decisions made in a centralized manner. There are some similarities between our work and that of [9]–[11] in that we exploit the robustness of subgradient-based sequential convex optimization schemes in the presence of perturbations. However, unlike these works, but similarly to [6], we use approximate dynamic programming to decentralize the decision-making process at each time step.

B. Notation

For any positive integer k, we will denote by $[k]$ the set $\{1, \ldots, k\}$. Given two real numbers a and b, we will write $a \wedge b$ for $\min\{a, b\}$ and $a \vee b$ for $\max\{a, b\}$. For any pair of vectors $x, y \in \mathbb{R}^d$, $\langle x, y \rangle$ will denote the standard Euclidean inner product, while $\|x\|$ will denote the ℓ_2 norm. All functions between Borel subsets of Euclidean spaces are assumed to be appropriately measurable. The space of all bounded measurable functions on a compact domain $\mathsf{X} \subset \mathbb{R}^d$ will be denoted by $M_b(\mathsf{X})$. For each $H \in M_b(\mathsf{X})$, we will denote by $\|H\|_\infty$ the usual sup norm and by $\|H\|_s$ the *span seminorm* [12]

$$\|H\|_s \triangleq \sup_{x \in \mathsf{X}} H(x) - \inf_{x \in \mathsf{X}} H(x).$$

We will use the notion of a subgradient [13]: $g \in \mathbb{R}^d$ is a subgradient of a convex function $f : \mathbb{R}^d \to \mathbb{R}$ at $x \in \mathbb{R}^d$ if

$$f(y) \geqslant f(x) + \langle g, y - x \rangle, \qquad \forall y \in \mathbb{R}^d.$$

The set of all subgradients of f at x will be denoted by $\partial f(x)$. Finally, for any two probability measures P, Q on a measurable space $(\Omega, \mathcal{B})$ the *total variation distance* is

$$\|P - Q\|_{TV} \triangleq 2 \sup_{B \in \mathcal{B}} |P(B) - Q(B)|.$$

II. The model

We consider decentralized sequential decision processes involving a large number n of agents, where the cost functions vary from stage to stage in an arbitrary manner and the agents' strategies are based on causally available local information (in a sense to be made precise shortly).

The centralized version of this problem was first posed and studied by Zinkevich [14] under the name of *Online Convex Programming* (OCP). It can be formulated as a repeated game between two players, the Agent and Nature. The moves of the Agent are points in a closed convex set U, while those of Nature are convex functions $f : \mathsf{U} \to \mathbb{R}$ in some class $\mathcal{F}$. At each round $t \in [T]$, the Agent plays a point $u_t \in \mathsf{U}$, Nature announces $f_t \in \mathcal{F}$, and the Agent incurs the cost $f_t(u_t) + \varphi(u_t)$, where $\varphi : \mathsf{U} \to \mathbb{R}$ is a fixed and known convex *regularization function*[2]. It is assumed that the Agent has *perfect recall* and can base his choice of u_t on all previous moves $u^{t-1} = (u_1, \ldots, u_{t-1})$ and all previous cost functions $f^{t-1} = (f_1, \ldots, f_{t-1})$. Thus, we can describe the Agent's *strategy* by a sequence $\gamma = \{\gamma_t\}_{t=1}^T$, where γ_t maps the information available to the Agent at time t to his move $u_t = \gamma_t(u^{t-1}, f^{t-1})$. The overall performance of the Agent is measured by the *regret*

$$R_T(\gamma, f^T) \triangleq \sum_{t=1}^T [f_t(u_t) + \varphi(u_t)] - \inf_{u \in \mathsf{U}} \sum_{t=1}^T [f_t(u) + \varphi(u)]$$

– the difference between the Agent's total cost and the smallest cost that could have been incurred in hindsight by using the best single move with full knowledge of f^T. The main quantity of interest is the *minimax regret*

$$R_T^*(\mathcal{F}) \triangleq \inf_\gamma \sup_{f^T \in \mathcal{F}^T} R_T(\gamma, f^T),$$

where the infimum is over all admissible strategies and the supremum is over all possible sequences of Nature's moves. The objective is to ensure that $R_T^*(\mathcal{F})$ is a *sublinear* function of the horizon T. When $\mathcal{F}$ is the class of all convex L-Lipschitz functions on U, a simple strategy based on projected subgradient descent guarantees that $R_T^*(\mathcal{F}) = O(\sqrt{T})$, where the constant hidden in the $O(\cdot)$ notation depends on the diameter of U and on L [14], [15]. Moreover, $R_T^*(\mathcal{F}) = \Omega(\sqrt{T})$ [16], so we have minimax optimality.

A. Decentralized online convex programming

In this work, we introduce a *decentralized* generalization of OCP involving a large number of agents, each of whom receives only *partial* information about the past cost functions and the past decisions of the other agents. Specifically, consider a team of n agents, $\mathrm{A}_1, \ldots, \mathrm{A}_n$. The decision of each agent is a scalar in $[0, 1]$.[3] The overall decision space

[2]Zinkevich considered the case $\varphi \equiv 0$; the more general regularized formulation has been analyzed recently by Xiao [15].

[3]The restriction of the individual decisions to $[0, 1]$ is not essential, and is imposed mainly for simplicity. Everything just as easily goes through if each agent makes decisions in a compact convex subset of $\mathbb{R}^d$.

is $\mathsf{U} = [0,1]^n$. We assume that both the cost and the regularization functions can be decomposed as

$$f(u) = f(u_1, \ldots, u_n) = \frac{1}{n} \sum_{i=1}^{n-1} f_i(u_i, u_{i+1}), \qquad (1)$$

$$\varphi(u) = \varphi(u_1, \ldots, u_n) = \frac{1}{n} \sum_{i=1}^{n-1} \varphi_i(u_i, u_{i+1}). \qquad (2)$$

Here, the f_i's are elements of a given class $\mathcal{C}$ of convex functions $c : [0,1]^2 \to \mathbb{R}$, while the φ_i's are fixed and known convex functions $[0,1]^2 \to \mathbb{R}$. The regularization terms $\{\varphi_i\}$ are assumed known to all agents. This is the simplest nontrivial cost structure, in which each agent's decision affects the performance of other agents, and where partial knowledge of the cost function is synonymous with knowing *some* (but not all) of the local terms $f_1, \ldots, f_{n-1}$. We will denote the set of all functions of the form (1) by $\mathcal{F}_n(\mathcal{C})$.

In order to describe the interaction of each agent with Nature and with other agents, we introduce the notion of an *information structure* in the spirit of Witsenhausen's framework for decentralized stochastic control [1]:

Definition 1. *An* information structure $\mathbf{I}$ *is a specification, for each* $i \in [n], t \in [T]$*, of the sets*

$$D_i^t \subseteq [n] \times [t-1], \qquad F_i^t \subseteq [n-1] \times [t-1].$$

The role of D_i^t (resp., F_i^t) is to determine which past decisions (resp., local cost functions) are visible to A_i at time t. The information structure $\mathbf{I}^*$ consisting of

$$D_i^t = [n] \times [t-1] \qquad \text{and} \qquad F_i^t = [n-1] \times [t-1]$$

for each t, i describes the centralized case.

We will consider strategies that allow the agents to make use of common randomness, which is known to be of help in decentralized scenarios. To that end, we assume that, for every $i \in [n]$, A_i generates a T-tuple $\{W_{i,t}\}_{t=1}^T$ of i.i.d. Uniform$[0,1]$ random variables independently of all other agents, and that if A_i and A_j share information at time t, then they also share $W_{i,t}$ and $W_{j,t}$. Then the move of A_i at time t will itself be a random variable, which we will denote by $U_{i,t}$. These random variables are specified recursively. Let us define the *information state* of A_i at time t by

$$I_i^t \triangleq \Big(U_{j,\tau}, (j,\tau) \in D_i^t; f_{k,\sigma}, (k,\sigma) \in F_i^t; W_{\ell,\nu}, (\ell,\nu) \in D_i^t \cup F_i^t\Big),$$

where $f_{k,\sigma} \in \mathcal{C}$ is the kth local term in the cost function $f_\sigma \in \mathcal{F}_n(\mathcal{C})$ selected by Nature at time σ. The information state captures all the data A_i can use at time t to form his decision. Given an information structure $\mathbf{I}$ and the corresponding set of information states I_i^t, a *strategy* for the team is a sequence $\gamma = \{\gamma_t\}_{t=1}^T$, where each $\gamma_t = (\gamma_{1,t}, \ldots, \gamma_{n,t})$ is an n-tuple of mappings, such that $U_{i,t} = \gamma_{i,t}\big(I_i^t\big)$ is the move of A_i at time t. Let $\Gamma(\mathbf{I})$ denote the space of all such strategies.

We can now formalize Decentralized OCP (or DOCP, for short) by means of the following protocol:

Decentralized Online Convex Programming (DOCP)

for $t = 1$ to T
  Nature chooses $f_t = (f_{1,t}, \ldots, f_{n-1,t}) \in \mathcal{F}_n(\mathcal{C})$
  for $i = 1$ to n
    A_i observes I_i^t and computes $U_{i,t} = \gamma_{i,t}\big(I_i^t\big)$
  end for
  the team incurs the cost $f_t(U_t) + \varphi(U_t)$
end for

Now, for each $\gamma \in \Gamma(\mathbf{I})$ and each choice of $f^T \in \mathcal{F}_n(\mathcal{C})^T$ we can define the *expected regret* $\bar{R}_T(\gamma, f^T)$ as

$$\mathbb{E}\left\{\sum_{t=1}^T f_t(U_t) + \varphi(U_t)\right\} - \inf_{u \in \mathsf{U}} \sum_{t=1}^T \left[f_t(u) + \varphi(u)\right],$$

where the expectation is w.r.t. $\{W_{i,t}\}$. The corresponding minimax regret now depends not only on the base class $\mathcal{C}$, but also on the underlying information structure $\mathbf{I}$:[4]

$$\bar{R}_T^*(\mathbf{I}, \mathcal{C}) \triangleq \inf_{\gamma \in \Gamma(\mathbf{I})} \sup_{f^T \in \mathcal{F}_n(\mathcal{C})^T} \bar{R}_T(\gamma, f^T)$$

III. Problem formulation and main result

Clearly, depending on how severely the prevailing information structure restricts the agents' capabilities, we may or may not be able to attain sublinear minimax regret. In this work, we will show that sublinear minimax regret is, indeed, attainable under a particular information structure inspired by the work of Rusmevichientong and van Roy [6]. Under this information structure, the agents are arranged on a chain according to their number, and each agent has perfect recall and can share information with all agents that are at most r steps away in either direction.

Definition 2. *For* $r \leqslant n$*, the* r-local information structure $\mathbf{I}^r$ *consists of*

$$D_i^t = \{(i-r) \vee 1, \ldots, (i+r) \wedge n\} \times [t-1]$$
$$F_i^t = \{(i-r) \vee 1, \ldots, (i+r) \wedge (n-1)\} \times [t-1]$$

Under $\mathbf{I}^r$, the influence of each agent is localized to a neighborhood of radius r. Such localization may be desirable for ensuring robustness to departures, arrivals, or failures of agents. Another desirable feature of a decentralized strategy, as explained in [6], is for any two agents to behave similarly when the cost- and decision-relevant parts of their information states are the same. Assuming that the number n of agents is very large (so boundary effects can be neglected), this essentially means that the agents will have to make decisions independently of their position in the chain:

Definition 3. *A strategy* $\gamma \in \Gamma(\mathbf{I}^r)$ *is* translation-invariant *if* $\gamma_{i,t} = \gamma_{j,t}$ *for all* $t \in [T]$ *and all* $i, j = r+1, \ldots, n-r$.

Our main result can be stated as follows:

[4] We are abusing the notation somewhat by writing $\mathcal{C}$ instead of $\mathcal{F}_n(\mathcal{C})$, but since it is $\mathcal{C}$ that determines the cost functions for the team, hopefully this will not lead to confusion.

Theorem 1. *Let $\mathcal{C}$ consist of all functions $c : [0,1]^2 \to \mathbb{R}$ that are convex and L-Lipschitz, i.e., $|c(x,y) - c(x',y')| \leqslant L\sqrt{(x-x')^2 + (y-y')^2}$ for all $x, x', y, y' \in [0,1]$, and let the local regularization functions $\varphi_i : [0,1]^2 \to \mathbb{R}$, $i \in [n]$, be convex and M_φ-Lipschitz. Then, provided*

$$r \geqslant \frac{3\log T + \log 2}{\log\left(\dfrac{2T^{3/2}}{2T^{3/2}-1}\right)}, \tag{3}$$

there exists a translation-invariant strategy $\gamma \in \Gamma(\mathbf{I}^r)$ with

$$\sup_{f^T \in \mathcal{F}_n(\mathcal{C})^T} \bar{R}_T(\gamma, f^T) = O(\sqrt{T}) \Rightarrow \bar{R}_T^*(\mathbf{I}^r, \mathcal{C}) = O(\sqrt{T})$$

The significance of this result is that the minimal communication radius needed to achieve the optimal $O(\sqrt{T})$ minimax regret is $\Omega(T^{3/2}\log T)$, *independently* of the number of agents n. This fact has deep implications for designing information structures in large networks when the typical planning horizon is known beforehand.

The rest of the paper is devoted to a constructive proof of the theorem. To that end, we first develop an explicit strategy in Sections IV and V, and then present the proof itself in Section VI. Although our results pertain to the line graph, the key concepts can be extended to more general network topologies (information structures); this extension is an important element of our ongoing work.

IV. The proposed solution

In a nutshell, we will take a deterministic strategy that attains minimax optimal regret in the centralized case, specifically the Regularized Dual Averaging (RDA) scheme of Xiao [15], and then develop a stochastic decentralized approximation utilizing the agents' shared randomness.

Let us first explain how RDA works in the centralized setting. We start by choosing a *proximal function* $\Phi : \mathsf{U} \to \mathbb{R}$, which is assumed to be differentiable and *strongly convex* with parameter $m_\Phi > 0$, i.e., for all $u, v \in \mathsf{U}$

$$\Phi(v) \geqslant \Phi(u) + \langle \nabla\Phi(u), v-u\rangle + \frac{m_\Phi}{2}\|u-v\|^2.$$

For example, if Φ is twice differentiable and all the eigenvalues of the Hessian $\nabla^2\Phi$ are at least m_Φ throughout U, then Φ is strongly convex. We also assume that $\Phi(u) \geqslant 0$ for all $u \in \mathsf{U}$ and that $\Phi(0) = 0$. The algorithm maintains two sequences in $\mathbb{R}^n$: the *primal* sequence $\{u_t\}_{t=0}^\infty$ and the *dual* sequence $\{z_t\}_{t=0}^\infty$, where $u_0 = z_0 = 0$, $z_{t+1} = z_t + g_{t+1}$ for $t = 0, 1, \ldots$ with $g_t \in \partial f_t(u_t)$ an arbitrary subgradient of f_t at u_t, and

$$u_{t+1} = \Pi_{\beta_t}^t(z_t), \qquad t = 0, 1, \ldots, \tag{4}$$

where $\{\beta_t\}_{t=0}^\infty \subset \mathbb{R}$ is a nondecreasing sequence with $\beta_0 = 1$, and for any $\beta > 0$, $z \in \mathbb{R}^n$, $u \in \mathsf{U}$,

$$\Pi_\beta^t(z) \triangleq \arg\min_{u\in\mathsf{U}} \underbrace{\{\langle z, u\rangle + t\varphi(u) + \beta\Phi(u)\}}_{\triangleq H_\beta^t(z,u)}$$

When the functions f_t are uniformly Lipschitz, the choice $\beta_t = \sqrt{t+1}$ leads to $O(\sqrt{T})$ regret [15, Section 3.1].

To adapt this algorithm to the decentralized setting, we will choose a proximal function Φ of the form

$$\Phi(u) = \frac{1}{n}\sum_{i=1}^{n-1} \Phi_i(u_i, u_{i+1}). \tag{5}$$

We will also assume that $\|\Phi\|_\infty = C_\Phi < \infty$. Let us write down the explicit form of the RDA updates when the cost functions f_t are elements of $\mathcal{F}_n(\mathcal{C})$ and Φ takes the form (5). For each $\tau = 1, \ldots, t$ let $g_{i,\tau} = \left(g_{i,\tau}^{(1)}, g_{i,\tau}^{(2)}\right) \in \mathbb{R}^2$ denote an arbitrary subgradient of $f_{i,\tau}$ at $(u_{i,\tau}, u_{i+1,\tau})$. Then let $\xi_t = (\xi_{1,t}, \ldots, \xi_{n,t}) \in \mathbb{R}^n$ be the vector with components

$$\xi_{i,t} = \begin{cases} \sum_{\tau=1}^t g_{1,\tau}^{(1)}, & i = 1 \\ \sum_{\tau=1}^t \left(g_{i,\tau}^{(1)} + g_{i-1,\tau}^{(2)}\right), & i = 2, \ldots, n-1 \\ \sum_{\tau=1}^t g_{n-1,\tau}^{(2)}, & i = n \end{cases} \tag{6}$$

It is not hard to show that $\xi_t = nz_t \equiv n\sum_{\tau=1}^t g_\tau$, where $g_\tau \in \partial f_\tau(u_\tau)$ for each τ. Hence, the computation of u_{t+1} in the centralized case entails minimizing the function

$$H_{\beta_t}^t(z_t, u) = \frac{1}{n}\sum_{i=1}^{n-1} h_{i,t}(u_i, u_{i+1}), \tag{7}$$

where $h_{i,t}(u_i, u_{i+1}) =$

$$\begin{cases} \xi_{i,t}u_i + t\varphi_i(u_i, u_{i+1}) + \beta_t\Phi_i(u_i, u_{i+1}), & 1 \leqslant i \leqslant n-2 \\ \xi_{n-1,t}u_{n-1} + \xi_{n,t}u_n + t\varphi_{n-1}(u_{n-1}, u_n) & \\ \qquad + \beta_t\Phi_{n-1}(u_{n-1}, u_n), & i = n-1 \end{cases} \tag{8}$$

Now let us consider the decentralized scenario with the r-local information structure. For each i, let us define

$$G_i = \{(i-r+1)\vee 1, \ldots, (i+r-1)\wedge(n-1)\}.$$

Suppose that the decisions $u_1, \ldots, u_t$ have already been made. Let us look at the problem faced by A_i at time t, namely, the computation of $u_{i,t+1}$. If all of a sudden A_i were to gain access to all past decisions and cost functions, then his best action would be to compute the ith component of $\Pi_{\beta_t}^t(z_t)$. However, under the r-local information structure, the information state I_i^{t+1} still permits the computation of $h_{j,t}$ for every $j \in G_i$ using (6) and (8). Thus, a reasonable strategy for A_i would be to try and approximate the ith component of $\Pi_{\beta_t}^t(z_t)$ using the local data $(\xi_{j,t} : j \in G_i)$. In other words, the overall team strategy is to *approximate* the RDA updates (4), where each agent would use only local information pertaining to the dual sequence $\{z_t\}$.

This local approximation will be facilitated by a procedure we will refer to as the *Local Dynamic Programming Relaxation (LDPR)*. This procedure, whose description and analysis are given in the next section, is an extension of the methods developed in [6] to decentralized Euclidean optimization problems of the form (7). For now, we represent it abstractly as a black box function with parameter $\delta \in (0,1)$, inputs $h_1, \ldots, h_{n-1}, W_1, \ldots, W_n$, and outputs $U_1, \ldots, U_n$:

$$(U_1, \ldots, U_n) \leftarrow \mathrm{LDPR}_\delta\left(\{h_i\}_{i=1}^{n-1}, \{W_j\}_{j=1}^n\right).$$

Then our proposed strategy will take the following form:

DOCP Under r-Local Information Structure

$U_{1,1} = U_{2,1} = \ldots = U_{n,1} = 0$
for $t = 1$ to $T-1$
  for $i = 1, \ldots, n$
    A_i observes I_i^{t+1} and computes $h_{j,t}, j \in G_i$
  end for
  $(U_{1,t+1}, \ldots, U_{n,t+1}) = \text{LDPR}_\delta\left(\{h_{i,t}\}_{i=1}^{n-1}, \{W_{j,t}\}_{j=1}^{n}\right)$
end for

V. Local DP relaxation: construction and analysis

The objective of LDPR is to minimize a cost function

$$h(u) = \frac{1}{n}\sum_{i=1}^{n-1} h_i(u_i, u_{i+1}) \tag{9}$$

using a team of n agents where, for each $i \in [n]$, only the terms h_j with $j \in G_i$ are revealed to A_i. We assume that the h_i's are continuous and uniformly bounded,

$$\max_{i=1,\ldots,n-1} \|h_i\|_\infty \leqslant B. \tag{10}$$

Our construction of LDPR, which builds on the methods developed in [6] in the context of combinatorial optimization over finite decision spaces, involves two steps: first, the deterministic minimization problem (9) is relaxed to a Markov decision process (MDP) with a specific transition law, and then the dynamic programming (DP) recursion for this MDP is approximated locally. Crucially, the common randomness available to the agents is needed to guarantee that the local approximations are close to the global optimum.

To construct the MDP relaxation, let $\mathsf{X} = \mathsf{A} = [0,1]$, where X will be the state space and A will be the action (control) space, choose some $\delta \in (0,1)$, and consider the transition kernel $P_\delta(dx'|x,a) = P_\delta(dx'|a)$, such that

$$x' = \begin{cases} a, & \text{with probability } 1-\delta \\ 0, & \text{with probability } \delta/2 \\ 1, & \text{with probability } \delta/2 \end{cases} \tag{11}$$

Let $h_0 \equiv 0$. Given a Markov policy $\pi = (\mu_0, \ldots, \mu_{n-1})$, where each μ_i is a (measurable) function from X into A, consider the total expected cost starting at $x_0 \in \mathsf{X}$:

$$C_{x_0}(\pi) \triangleq \frac{1}{n}\mathbb{E}_{x_0}^{\pi}\left\{\sum_{i=0}^{n-1} h_i(X_i, \mu_i(X_i))\right\},$$

where $\mathbb{E}_{x_0}^{\pi}\{\cdot\}$ is the expectation w.r.t. the probability measure induced by π starting at $X_0 = x_0$. The connection between this MDP and mininization of h is revealed by the following lemma, whose easy proof we omit:

Lemma 1. *For any policy π and any initial $x_0 \in \mathsf{X}$,*

$$\frac{1}{n}\mathbb{E}_{x_0}^{\pi}\left\{\sum_{i=1}^{n-1} h_i(X_i, X_{i+1})\right\} \leqslant C_{x_0}(\pi) + 2B\delta$$

$$\frac{1}{n}\inf_{\pi}\mathbb{E}_{x_0}^{\pi}\left\{\sum_{i=1}^{n-1} h_i(X_i, X_{i+1})\right\} \leqslant \inf_{u\in\mathsf{U}} h(u) + 4B\delta.$$

In the centralized setting, the optimal policy π^* that minimizes $C_{x_0}(\pi)$ for all x_0 can be computed using DP. We now show how each agent can implement an approximate DP recursion based only on locally available information. To that end, we first define the standard mappings associated with the DP recursion [17, Chap. 3]: For every $i = 0, 1, \ldots, n-1$ and $\mu : \mathsf{X} \to \mathsf{A}$, define the operators T^i and T_μ^i that map any $H : \mathsf{X} \to \mathbb{R}$ to $T^iH, T_\mu^iH : \mathsf{X} \to \mathbb{R}$ via

$$(T^iH)(x) \triangleq \inf_{a\in\mathsf{A}}\left\{\frac{1}{n}h_i(x,a) + \mathbb{E}_\delta\left\{H(X')|a\right\}\right\}$$

$$(T_\mu^iH)(x) \triangleq \frac{1}{n}h_i(x,\mu(x)) + \mathbb{E}_\delta\left\{H(X')|\mu(x)\right\}.$$

From the vantage point of A_i who only has access to $h_{i-r+1}, \ldots, h_{i+r-1}$, the cost-to-go given the state $X_{i-1} = x_{i-1}$ and any policy $\pi = (\mu_0, \ldots, \mu_{n-1})$ is

$$C_{x_{i-1}}^{(i)}(\pi) = \mathbb{E}^{\pi}\left\{\sum_{j=i-1}^{i+r-1} h_j(X_j, \mu_j(X_j))\,\middle|\, X_{i-1} = x_{i-1}\right\}.$$

The past costs visible to A_i are $h_{i-r+1}, \ldots, h_{i-1}$. Hence, A_i can implement the DP recursion starting at $j = i+r-1$ and descending to $j = i$,[5]

$$J_{i,r} \equiv 0; \quad J_{i,\ell} = T^{i+\ell}J_{i,\ell+1}, \ell = 0, \ldots, r-1$$

and for $\ell = -1, \ldots, r-1$ compute $\mu_{i+\ell}^{(i)} : \mathsf{X} \to \mathsf{A}$, such that

$$T_{\mu_{i+\ell}^{(i)}}^{i+\ell} J_{i,\ell+1} = T^{i+\ell}J_{i,\ell+1}. \tag{12}$$

Let $\mu_{i-1} \triangleq \mu_{i-1}^{(i)}$. Since A_i knows $h_{i-r+1}, \ldots, h_{i-2}$, he can also compute the mappings $\mu_{i-r+1}, \ldots, \mu_{i-2}$. Next, we bring in common randomness. Let $\{W_j\}_{j=1}^n$ be n i.i.d. Uniform$[0,1]$ random variables, where W_j is held by A_j. Since A_i has access to $(W_j : j \in G_i)$, he simply *simulates* the controlled Markov chain from $j = i-r+1$ to $j = i$ starting with the zero initial state and using the truncated policy $(\mu_{i-r+1}, \ldots, \mu_{i-1})$. For $W \sim$ Uniform$[0,1]$, let $F_\delta(a, W)$ denote the deterministic realization of the stochastic kernel $P_\delta(dx'|a)$. We can now summarize the entire method:

Local Dynamic Programming Relaxation (LDPR)

Parameter: $\delta \in (0,1)$
Input: $h_1, \ldots, h_{n-1}; W_1, \ldots, W_n$
Output: $(U_1, \ldots, U_n) = \text{LDPR}_\delta\left(\{h_i\}_{i=1}^{n-1}; \{W_j\}_{j=1}^{n}\right)$

for $i = 1, ..., n$
  A_i does the following:
    observe $(h_j, W_j : j \in G_i)$
    compute $\mu_j, j \in G_i \cap \{i-r+1, \ldots, i-1\}$
    let $\ell_0 = 0 \vee (i-r+1)$, $X_{\ell_0}^{(i)} = 0$
    for $\ell = \ell_0, \ldots, i-1$
      $X_{\ell+1}^{(i)} = F_\delta\Big(\mu_\ell\big(X_\ell^{(i)}\big), W_{\ell+1}\Big)$
    end for
    output $U_i = X_i^{(i)}$
end for

[5] To handle the case $i + r > n$, we can simply pad the cost functions $h_1, \ldots, h_{n-1}$ with $h_n = \ldots = h_{n+r-1} = 0$.

Note that LDPR is translation-invariant by construction, since any two agents facing the same sequence of local cost functions will implement the same DP recursion and thus will compute the same truncated policies.

A. Analysis of LDPR performance

Theorem 2. *Consider any tuple $h_1, \dots, h_{n-1}$ of continuous functions satisfying* (10)*. Then*

$$\frac{1}{n}\mathbb{E}\left\{\sum_{i=1}^{n-1} h_i(U_i, U_{i+1})\right\} - \inf_{u\in\mathsf{U}} h(u) \leqslant B\Delta(\delta, r),$$

where $(U_1, \dots, U_n)$ is the output of LDPR, and $\Delta(\delta, r) \triangleq 6\left(\delta + \left(1+\frac{1}{\delta}\right)\left(1-\frac{\delta}{2}\right)^r\right)$.

Proof. The proof essentially follows the same steps as in [6], except for a few modifications that arise because of working with continuous state and action spaces.

Let $\pi^* = (\mu_0^*, \dots, \mu_{n-1}^*)$ be the optimal centralized policy, and let $h^* = \inf_{u\in\mathsf{U}} h(u)$. Let $\pi = (\mu_0, \dots, \mu_{n-1})$ be the policy consisting of the mappings constructed by the n agents using LDPR and their local information. By Lemma 1,

$$\frac{1}{n}\mathbb{E}_0^\pi\left\{\sum_{i=1}^{n-1} h_i(X_i, X_{i+1})\right\} - h^* \leqslant C_0(\pi) - C_0(\pi^*) + 6B\delta.$$

The expected costs of π^* and π can both be expressed as

$$C_0(\pi^*) = \left[T_{\mu_0^*}^0 T_{\mu_1^*}^1 \dots T_{\mu_{n-1}^*}^{n-1} J_n^*\right](0)$$
$$C_0(\pi) = \left[T_{\mu_0}^0 T_{\mu_1}^1 \dots T_{\mu_{n-1}}^{n-1} J_n\right](0),$$

where $J_n^* = J_n = 0$, and the respective DP recursions are

$$J_i^* = T_{\mu_i^*}^i J_{i+1}^*,\ J_i = T_{\mu_i}^i J_{i+1} \qquad i = 0, \dots, n-1$$

Therefore, for any i we have

$$\begin{aligned}\|J_i^* - J_i\|_\infty &= \left\|T_{\mu_i^*}^i J_{i+1}^* - T_{\mu_i}^i J_{i+1}\right\|_\infty \\ &\leqslant \left\|T_{\mu_i^*}^i J_{i+1}^* - T_{\mu_i}^i J_{i+1}^*\right\|_\infty + \left\|J_{i+1}^* - J_{i+1}\right\|_\infty,\end{aligned}$$

where we have used the triangle inequality and the fact that T_μ^i is a contraction in the $\|\cdot\|_\infty$ norm [17]. From (12), $T_{\mu_i}^i J_{i+1,0} = T^i J_{i+1,0}$, so by Lemma A.1 in the Appendix,

$$\left\|T_{\mu_i^*}^i J_{i+1}^* - T_{\mu_i}^i J_{i+1}^*\right\|_\infty \leqslant \left\|J_{i+1}^* - J_{i+1,0}\right\|_s.$$

Using Lemma A.2 repeatedly and then Lemma A.3, we get

$$\|J_{i+1}^* - J_{i+1,0}\|_s \leqslant \frac{2B}{\delta n}\left(1-\frac{\delta}{2}\right)^r.$$

Since $J_n^* = J_n = 0$, we finally get

$$|C_0(\pi^*) - C_0(\pi)| \leqslant \|T^0 J_0^* - T^0 J_0\|_\infty \leqslant \frac{2B}{\delta}\left(1-\frac{\delta}{2}\right)^r$$

and therefore

$$\frac{1}{n}\mathbb{E}_0^\pi\left\{\sum_{i=1}^{n-1} h_i(X_i, X_{i+1})\right\} - h^* \leqslant 6B\delta + \frac{2B}{\delta}\left(1-\frac{\delta}{2}\right)^r.$$

Let P_i denote the probability law of (X_i, X_{i+1}) under policy π and $X_0 = 0$, and let Q_i denote the probability law of (U_i, U_{i+1}). Then, since $\|h_i\|_\infty \leqslant B$, we have

$$\begin{aligned}|\mathbb{E}_0^\pi\{h_i(X_i, X_{i+1})\} - \mathbb{E}\{h_i(U_i, U_{i+1})\}| &\leqslant B\|P_i - Q_i\|_{TV} \\ &\leqslant 4B(1-\delta)^r,\end{aligned}$$

where the last step uses Lemma A.4. Putting everything together, we get the desired bound. □

VI. Proof of Theorem 1

To prove Theorem 1, we will show that if the LDPR-based strategy is used with $\delta = 1/T^{3/2}$ and if the communication radius r satisfies (3), then we can guarantee $\bar{R}_T^*(\mathbf{I}^r, \mathcal{C}) = O(\sqrt{T})$. From this point on, $\gamma \in \Gamma(\mathbf{I}^r)$ will denote the LDPR-based strategy. Note that γ is indeed translation-invariant because LDPR is.

Given the sequence of decisions $\{U_t\}_{t=1}^T$, let us define another sequence $\{\bar{U}_t\}_{t=1}^T$ by $\bar{U}_1 = 0$ and

$$\bar{U}_{t+1} = \Pi_{\beta_t}^t(z_t) = \arg\min_{u\in\mathsf{U}}\{\langle z_t, u\rangle + t\varphi(u) + \beta_t\Phi(u)\}$$

for $t \geqslant 1$, where the z_t's are computed from $\{g_t\}$ according to (6) using the relation $z_t = (1/n)\xi_t$. Let $u^* \in \mathsf{U}$ be a minimizer of $\sum_{t=1}^T [f_t + \varphi]$. Then

$$\begin{aligned}R_T(\gamma, f^T) &= \sum_{t=1}^T [f_t(U_t) - f_t(u^*)] + \sum_{t=1}^T [\varphi(U_t) - \varphi(u^*)] \\ &\leqslant \underbrace{\sum_{t=1}^T \left[\langle g_t, \bar{U}_t - u^*\rangle + \varphi(\bar{U}_t) - \varphi(u^*)\right]}_{\text{(a)}} + \underbrace{\sum_{t=1}^T K\|U_t - \bar{U}_t\|}_{\text{(b)}}\end{aligned}$$

with $K = L + 2M_\varphi$, where the second step uses convexity of the f_t's and Lipschitz continuity of the f_t's and φ. This shows that the regret is bounded by the sum of two terms, (a) and (b), where term (a) is an optimization term and term (b) is the additional loss due to decentralization. We now analyze these two terms.

A. Optimization term

To tackle term (a), we use Proposition 2 in [10]:

Lemma 2. *Let $\{g_t\}_{t=0}^\infty \subset \mathbb{R}^n$ be an arbitrary sequence of vectors, let $\{\beta_t\}_{t=0}^\infty$ be a nondecreasing sequence of positive reals, and consider the sequence $\{\bar{u}_t\}_{t=1}^\infty$ defined by*

$$\bar{u}_{t+1} = \arg\min_{u\in\mathsf{U}}\left\{\sum_{\tau=0}^t \langle g_\tau, u\rangle + t\varphi(u) + \beta_t\Phi(u)\right\}.$$

Then for any $u \in \mathsf{U}$ we have

$$\sum_{t=1}^T [\langle g_t, \bar{u}_t - u\rangle + \varphi(\bar{u}_t) - \varphi(u)] \leqslant \sum_{t=1}^T \frac{\|g_t\|^2}{2\beta_{t-1}} + \beta_T\Phi(u).$$

With $g_0 = 0$, we can apply the lemma to $\{\bar{U}_t\}_{t=1}^T$ to get

$$\begin{aligned}&\mathbb{E}\left\{\sum_{t=1}^T \langle g_t, \bar{U}_t - u^*\rangle + \varphi(\bar{U}_t) - \varphi(u^*)\right\} \\ &\leqslant \sum_{t=1}^T \frac{\mathbb{E}\|g_t\|^2}{2\beta_{t-1}} + \beta_T\Phi(u^*) \leqslant \frac{L^2}{2}\sum_{t=1}^T \frac{1}{\beta_{t-1}} + C_\Phi\beta_T. \quad (13)\end{aligned}$$

B. Loss due to decentralization

Recall that U_{t+1} is an approximation to $\Pi^t_{\beta_t}(z_t)$, which is computed from $h_{1,t}, \dots, h_{n-1,t}$ using LDPR. On the other hand, $\bar{U}_{t+1}$ is the *exact* minimizer of $H^t_{\beta_t}(z_t, u)$ over $u \in \mathsf{U}$. Now, $H^t_{\beta_t}$ is a sum of a convex function and a strongly convex function, and so is itself strictly convex with constant $m_\Phi \beta_t$. Therefore, since $\bar{U}_{t+1}$ minimizes $H^t_{\beta_t}(z_t, u)$, we have

$$H^t_{\beta_t}(z_t, U_{t+1}) - H^t_{\beta_t}(z_t, \bar{U}_{t+1}) \geqslant \frac{m_\Phi \beta_t}{2} \|U_{t+1} - \bar{U}_{t+1}\|^2$$

(cf. Theorem 2.1.8 in [13]), which gives $\|U_{t+1} - \bar{U}_{t+1}\|$

$$\leqslant \sqrt{\frac{2}{m_\Phi \beta_t} \left[H^t_{\beta_t}(z_t, U_{t+1}) - \inf_{u \in \mathsf{U}} H^t_{\beta_t}(z_t, u) \right]}.$$

To bound the quantity under the square root, we use the fact that $U_{t+1} = \text{LDPR}_\delta(\{h_{i,t}\}_{i=1}^{n-1}, \{W_{j,t}\}_{j=1}^n)$ and appeal to Theorem 2. Specifically,

$$H^t_{\beta_t}(z_t, U_{t+1}) = \frac{1}{n} \sum_{i=1}^{n-1} h_{i,t}(U_{i,t+1}, U_{i+1,t+1}),$$

where the terms $h_{i,t}$ are of the form (8). A simple calculation shows that $\|h_{i,t}\|_\infty \leqslant (4L + M_\varphi)t + C_\Phi \beta_t \equiv B_t$. Consequently, $\mathbb{E}\|U_{t+1} - \bar{U}_{t+1}\|$

$$\leqslant \sqrt{\frac{2}{m_\Phi \beta_t} \mathbb{E}\Big\{ H^t_{\beta_t}(z_t, U_{t+1}) - \inf_{u \in \mathsf{U}} H^t_{\beta_t}(z_t, u) \Big\}}$$
$$\leqslant \sqrt{\frac{(8L + 2M_\varphi)t + 2C_\Phi \beta_t}{m_\Phi \beta_t} \Delta(\delta, r)}, \qquad (14)$$

where the first step uses Jensen's inequality, while the second uses Theorem 2.

C. Obtaining $O(\sqrt{T})$ regret

Putting together (13) and (14), we get $\bar{R}_T(\gamma, f^T)$

$$\leqslant K' \left\{ \sqrt{\Delta(\delta, r)} \sum_{t=1}^T \sqrt{\frac{t + \beta_t}{\beta_t}} + \sum_{t=1}^T \frac{1}{\beta_{t-1}} + \beta_T \right\}, \qquad (15)$$

where $K' = K'(L, \varphi, \Phi) > 0$ is a constant that depends only on L, M_φ, m_Φ, and C_Φ. Taking $\beta_t = \sqrt{t+1}$, we get

$$\sum_{t=1}^T \sqrt{\frac{t + \beta_t}{\beta_t}} = O(T^{5/4}) \quad \text{and} \quad \sum_{t=1}^T \frac{1}{\beta_{t-1}} + \beta_T = O(\sqrt{T}).$$

Hence, if we choose δ and r to arrange for $\Delta(\delta, r) = O(T^{-3/2})$, we will obtain $O(\sqrt{T})$ regret. Simple algebra shows that if $\delta = 1/T^{3/2}$ and r satisfies (3), then $\Delta(\delta, r) = O(T^{-3/2})$, and so we indeed obtain $O(\sqrt{T})$ regret.

APPENDIX

This appendix contains some technical lemmas needed in the proof of Theorem 2. The proofs of Lemmas A.1–A.3 follow essentially the same steps as in [6]; the statement and the proof of Lemma A.4 are different from those in [6]. Details are omitted for lack of space.

Lemma A.1. *For any two $H_1, H_2 \in M_b(\mathsf{X})$, let $\mu_1, \mu_2 : \mathsf{X} \to \mathsf{A}$ satisfy $T^i_{\mu_1} H_1 = T^i H_1$ and $T^i_{\mu_2} H_2 = T^i H_2$. Then $\|T^i_{\mu_1} H_1 - T^i_{\mu_2} H_1\|_\infty \leqslant \|H_1 - H_2\|_s$.*

Lemma A.2. *For any two $H_1, H_2 \in M_b(\mathsf{X})$,*

$$\|T^i H_1 - T^i H_2\|_s \leqslant \left(1 - \frac{\delta}{2}\right) \|H_1 - H_2\|_s.$$

Lemma A.3. *For each i, $\|J^*_i\|_s \leqslant \frac{2B}{\delta n}$.*

Lemma A.4. *For each i, let P_i denote the probability law of the couple (X_i, X_{i+1}) under the policy π and the initial state $X_0 = 0$, and let Q_i denote the probability law of the couple (U_i, U_{i+1}). Then $\|P_i - Q_i\|_{TV} \leqslant 4(1-\delta)^r$.*

REFERENCES

[1] H. S. Witsenhausen, "Separation of estimation and control for discrete-time systems," *Proc. IEEE*, vol. 59, no. 11, pp. 1557–1566, November 1971.

[2] A. Mahajan and S. Tatikonda, "Sequential team form and its simplification using graphical models," in *Proc. 47th Annual Allerton Conf. on Commun., Control and Computing*, 2009, pp. 330–339.

[3] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning and Games*. Cambrdige Univ. Press, 2006.

[4] M. Raginsky, A. Rakhlin, and S. Yüksel, "Online convex programming and regularization in adaptive control," in *Proc. IEEE Conf. on Decision and Control*, Atlanta, GA, December 2010, pp. 1957–1962.

[5] S. Tatikonda, "Control under communication constraints," Ph.D. dissertation, Dept. Elect. Eng. Comp. Sci., Massachusetts Institute of Technology, Cambridge, 2000.

[6] P. Rusmevichientong and B. van Roy, "Decentralized decision-making in a large team with local information," *Games and Economic Behavior*, vol. 43, no. 2, pp. 266–295, 2003.

[7] J. N. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Dept. Elect. Eng. Comp. Sci., Massachusetts Institute of Technology, Cambridge, 1984.

[8] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Automat. Control*, vol. AC-31, no. 9, pp. 803–812, 1986.

[9] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Automat. Control*, vol. 54, no. 1, pp. 48–61, 2009.

[10] J. Duchi, A. Agarwal, and M. Wainwright, "Dual averaging for distributed optimization: convergence analysis and network scaling," 2010, submitted. [Online]. Available: http://arxiv.org/abs/1005.2012

[11] F. Yan, S. Sundaram, S. V. N. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: regrets and intrinsic privacy-preserving properties," 2011. [Online]. Available: http://arxiv.org/abs/1006.4039

[12] O. Hernández-Lerma and J. B. Lasserre, *Discrete-Time Markov Control Processes: Basic Optimality Criteria*. Springer, 1996.

[13] Y. Nesterov, *Introductory Lectures on Convex Optimization*. Kluwer Academic Publishers, 2004.

[14] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. Int. Conf. on Machine Learning (ICML)*, 2003, pp. 928–936.

[15] L. Xiao, "Dual averaging methods for regulairzed stochastic learning and online optimization," *J. Machine Learn. Res.*, vol. 11, pp. 2543–2596, 2010.

[16] J. Abernethy, P. Bartlett, A. Rakhlin, and A. Tewari, "Optimal strategies and minimax lower bounds for online convex games," in *Proc. Conf. on Learning Theory (COLT)*, 2008, pp. 415–423.

[17] D. P. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete Time Case*. New York: Academic Press, 1978.