Distributed Mirror Descent Algorithm With Bregman Damping for Nonsmooth Constrained Optimization

Guanpu Chen, Gehui Xu, Weijian Li, and Yiguang Hong, Fellow, IEEE

Abstract—To efficiently solve the nonsmooth distributed optimization with both local constraints and coupled constraints, we propose a distributed continuous-time algorithm based on the mirror descent (MD) method. In this article, we introduce the Bregman damping into distributed MD-based dynamics, which not only successfully applies the MD idea to the distributed primal-dual framework, but also ensures the boundedness of all variables and the convergence of the entire dynamics. Our approach generalizes the classic distributed projection-based dynamics, and establishes a connection between MD methods and distributed Euclidean-projected approaches. Also, we prove the convergence of the proposed distributed dynamics with an $O(1/t)$ rate. For practical implementation, we further give a discrete-time algorithm based on the proposed dynamics with an $O(1/\sqrt{t})$ convergence rate.

Index Terms—Constrained optimization, distributed algorithm, mirror descent, multi-agent system, nonsmooth.

I. INTRODUCTION

Distributed optimization serves as a hot topic in recent years for its broad applications in various fields, such as sensor networks and smart grids [1], [2], [3], [4], [5], [6], [7]. Under multiagent frameworks, the global cost function consists of local ones, and each agent shares limited information with its neighbors through networks to achieve an optimal solution. Thanks to system dynamics and control theory, distributed continuous-time algorithms have been well developed [8], [9], [10], [11], [12].

Particularly, distributed constrained optimization is a more complicated but interesting research problem. Given the diversity of different constraints, various distributed methods have been exploited. As for local constraints, implementing projection operations in algorithms is the most popular method, such as projected proportional-integral protocol [8] and projected dynamics with constraints based on KKT conditions [9]. Moreover, the coupled constraint is another challenging structure in distributed optimization, which has been widely considered in resource allocation [13], [14], [15]. The primal-dual framework coordinating with decoupling techniques serves as a popular methodology in such situations [10], [11], [16]. Nevertheless, the approaches mentioned above are suitable for general distributed constrained optimization, rather than paying enough attention to the specific structures within the various constraints. Especially, time complexity in finding optimal solutions with complex or high-dimensional constraints forces researchers to exploit efficient approaches for special constraint structures, such as the unit simplex and the Euclidean sphere.

In fact, the mirror descent (MD) method is a well-known tool to overcome the bottleneck above. As we know, first introduced in [17], the MD is regarded as a generalization of (sub)gradient methods. By mapping the variables into a conjugate space, the MD employs the Bregman divergence and performs well in handling the constraints with specific structures [18], [19], [20]. This process results in a faster convergence rate than projected (sub)gradient algorithms with respect to problem dimensions [21], and is suitable for solving large-scale optimization problems. Undoubtedly, as such an important tool, the MD has played a crucial role in distributed algorithm design, as given in [22], [23], [24], and [25].

The objective of this article is to design an MD-based distributed continuous-time algorithm for nonsmooth optimization endowed with both local and coupled constraints. We intend to carry forward under the primal-dual framework, considering the importance of both distributed constrained optimization and powerful MD methods for handling specific constraint structures. In the following, we will address the contributions and related work.

Contributions:
1) We propose a distributed continuous-time MD algorithm to solve nonsmooth optimization problems with both local constraints and coupled constraints [4], [26], [27]. Our algorithm demonstrates its effectiveness to handle such a nonsmooth constrained problem, and well inherits the good capabilities of MD-based methods to rapidly compute explicit solutions for specific constraint structures.
2) By introducing the Bregman damping term in the distributed dynamics, we not only successfully apply the MD to handle coupled constraints, but also ensure the boundedness of all variables when facing compact local constraints [18], [24], [28]. Also, by taking quadratic forms into the Bregman damping, our MD-based approach can actually derive the classic distributed Euclidean-projected dynamics [8], [16], which establishes a connection between MD methods and distributed projection-based continuous-time algorithms.
3) By virtue of nonsmooth techniques, conjugate functions, and the Lyapunov theory, we prove the convergence of the MD-based dynamics with an $O(1/t)$ rate. Meanwhile, for practical implementation, we further provide a discrete version of the proposed dynamics. We investigate the average-iteration convergence rate of
the discrete algorithm, and obtain an $O(1/\sqrt{\mathcal{E}})$ convergence rate, which is of the same order as some existing discrete MD-based results [21], [29].

Related work:

1) **Coupled constraint:** Of particular relevance to this work is the literature on distributed problems with coupled constraints, which have recently drawn a great deal of attention and had broad application in resource allocation tasks. For coupled equality constraints, most related works usually focus on affine issues to restrict the scope of convexity, such as the distributed designs in [4], [13], and [16]. For coupled inequality constraints, the decoupling process is usually more complicated but diverse. For instance, Chang et al. [14] employed an average consensus technique for a primal-dual subgradient method, while Liang et al. [15] employed a projected form of singular perturbation for suboptimal solutions. More interestingly, the authors in [30] and [27] investigated distributed generalized Nash equilibria in game problems with coupled inequality constraints, via an operator splitting approach and approximate inscribed polyhedrons, respectively. To the best of the authors’ knowledge, the investigation of a distributed problem with both coupled equality constraints and inequality ones from the perspective of the MD methodology, which is more general than projection-based approaches, is still an open problem.

2) **Mirror descent:** Another highly relevant topic to this study is the MD method. As an important generalization of subgradient descent methods on non-Euclidean geometries, the MD method was first introduced by Nemirovskii and Yudin [17]. Since then, its implementation has been widely discussed in various applications, such as stochastic optimization [31], distributed optimization [22], [29], and online learning [32]. Most of the above were considered in discrete-time algorithms, and accordingly, the continuous-time MD dynamics have also been developed gradually, and the convergence has been of major interest to the control and optimization community. For example, Krichene et al. [18] proposed the acceleration of a continuous-time MD algorithm with an $O(1/t^2)$ convergence rate, while Gao and Pavel [28] proposed a discounted continuous-time MD dynamics to obtain approximate solutions. In the distributed design, Sun and Shahrampour [24] presented a distributed MD dynamics with integral feedback, whose result was only embodied in the unconstrained problem. Afterward, Sun and Shahrampour [33] extended this distributed dynamics on a class of strongly convex problems, and achieved a local exponential convergence rate. Additionally, Yu and Aşkınmeşe [26] motivated MD using RLC to reach an $O(1/k)$ rate of discretization with certain smoothness.

To sum up, dealing with both local constraints and coupled constraints under the distributed nonsmooth primal-dual framework is actually beyond the abovementioned discussions. Novel techniques are necessary to guarantee the dynamics avoiding unbounded variables [18], [24] and inaccurate convergence [28], and to provide an acceptable convergence rate.

II. PRELIMINARY

In this section, we give necessary notations and related preliminary knowledge.

**Notations:** Denote $\mathbb{R}^n$ (or $\mathbb{R}^{m \times n}$) as the set of $n$-dimensional (or $m$-by-$n$) real column vectors (or real matrices), and $I_n$ as the $n \times n$ identity matrix. Let $I_m$ (or $0_m$) be the $n$-dimensional column vector with all entries of 1 (or 0). Denote $A \oplus B$ as the Kronecker product of matrices $A$ and $B$. Take $\text{col}\{x_1, \ldots, x_n\} = \text{col}\{x_i\}_{i=1}^n = (x_1^T, \ldots, x_n^T)^T$, $\|\cdot\|$ as the Euclidean norm, and $\text{rint}(C)$ as the relative interior of the set $C$ [34].

**Graph theory:** An undirected graph can be defined by $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, N\}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. Let $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ be the adjacency matrix of $\mathcal{G}$ such that $a_{ij} = a_{ji} > 0$ if $\{i, j\} \in \mathcal{E}$, and $a_{ij} = 0$, otherwise. The Laplacian matrix is $L_N = D - A$, where $D = \text{diag}(D_{ii}) \in \mathbb{R}^{N \times N}$ with $D_{ii} = \sum_{j=1}^n a_{ij}$. If the graph $\mathcal{G}$ is connected, then $\ker(L_N) = \{k1_N : k \in \mathbb{R}\}$.

**Convex analysis:** For a closed convex set $\Omega \subseteq \mathbb{R}^n$, the projection map $P_\Omega : \mathbb{R}^n \to \Omega$ is defined as $P_\Omega(x) = \arg \text{min}_{y \in \Omega} \|x - y\|$. Especially, denote $[x]^+ = P_{\mathbb{R}_+}(x)$ for convenience. For $x \in \Omega$, denote the normal cone to $\Omega$ at $x$ by $N_\Omega(x) = \{v \in \mathbb{R}^n : v^T(y - x) \leq 0, \forall y \in \Omega\}$. A continuous function $f : \mathbb{R}^n \to \mathbb{R}$ is $\omega$-strongly convex on $\Omega$ if $(x - y)^T(g_x - g_y) \geq \omega\|x - y\|^2$, $\forall x, y \in C$, where $g_x \in \partial f(x)$ and $g_y \in \partial f(y)$.

The Bregman divergence based on the differentiable generating function $\phi : \mathbb{R} \to \mathbb{R}$ is defined as

$$D_\phi(x, x') = \phi(x) - \phi(x') - \nabla \phi(x')^T(x - x') \quad \forall x, x' \in \Omega.$$ 

The convex conjugate function of $\phi$ is defined as $\phi^*(y) = \sup_{x \in \Omega} \{ -x^T y + \phi(x) \}$. The following lemma reveals a classical conclusion about convex conjugate functions, of which readers can find more details in [17] and [19].

**Lemma 1:** Take function $\phi$ differentiable and strongly convex on a closed convex set $\Omega$. Then, $\phi^*(y)$ is convex and differentiable, and $\phi^*(y) = \min_{x \in \Omega} \{ -x^T y + \phi(x) \}$. Moreover, the mirror map satisfies $\nabla \phi^*(y) = \Pi_{\Omega}^\phi(y)$, where

$$\Pi_{\Omega}^\phi(y) = \arg \min_{x \in \Omega} \{-x^T y + \phi(x)\}. \tag{1}$$

**Differential inclusion:** A differential inclusion is given by

$$\dot{x}(t) = F(x(t)), x(0) = x_0, t \geq 0,$$

where $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is a set-valued map. $F$ is upper semicontinuous at $x$ if there exists $\delta > 0$ for given $\epsilon > 0$ such that $F(y) \subseteq F(x) + B(x; \delta), \forall y \in B(x; \delta)$. The existence of solutions to the abovementioned differential inclusion is guaranteed by the following lemma [35].

**Lemma 2:** If $F$ is locally bounded, upper semicontinuous, compact, and convex, then there exists a Carathéodory solution for any initial value.

Moreover, let $V$ be a locally Lipschitz continuous function. By [35], the set-valued derivative for $V$ is defined by $L_xV(x) = \{a \in \mathbb{R} : a = p^TV, p \in \partial V(x), v \in F(x)\}$.

III. FORMULATION

In this article, we consider a nonsmooth optimization problem with both local and coupled constraints. There are $N$ agents indexed by $\mathcal{V} = \{1, \ldots, N\}$ in a network $\mathcal{G}(\mathcal{V}, \mathcal{E})$. For agent $i$, the decision variable is $x_i$, the local feasible set is $\Omega_i \subseteq \mathbb{R}^n$, and the local cost function is $f_i : \mathbb{R}^n \rightrightarrows \mathbb{R}$. Define $\Omega = \prod_{i=1}^N \Omega_i$, and $x = \text{col}\{x_i\}_{i=1}^N$. All agents cooperate to solve the following distributed optimization:

$$\min_{x \in \Omega} \sum_{i=1}^N f_i(x_i),$$

s.t. $\sum_{i=1}^N g_i(x_i) \leq 0$, $\sum_{i=1}^N A_ix_i - b_i = 0$, \tag{2}$

where $g_i : \mathbb{R}^n \rightrightarrows \mathbb{R}^p$, $A_i \in \mathbb{R}^{m \times n}$, and $b_i \in \mathbb{R}^n$, for $i \in \mathcal{V}$. Except for the local constraint $\Omega_i$, other constraints in (2) are said to be coupled.
TABLE I
MIRROR MAP WITH DIFFERENT GENERATING FUNCTIONS

| Feasible set                     | Generating function $\phi(x)$ | Mirror map $\Pi_\Omega^\phi (y)$ |
|---------------------------------|--------------------------------|---------------------------------|
| General closed and convex set   | $\Omega$                      | $\frac{1}{2} \|x\|^2$          | $P_\Omega(y) = \arg\min_{x \in \Omega} \frac{1}{2} \|x - y\|^2$ |
| Non-negative orthant            | $\mathbb{R}^n_+$              | $\sum_{i=1}^{n} x_i \log(x_i) - x_i$ | $\exp(y)$ |
| Unit square $[a, b]^n$          | $\{ x \in \mathbb{R}^n : a \leq x_i \leq b \}$ | $\sum_{i=1}^{n} (x_i - a) \log(x_i - a) + (b - x_i) \log(b - x_i)$ | $\co\{ \frac{a + \exp(\|y\|_2^2)}{\exp(y)} \}_i=1$ |
| Simplex $\Delta^n$             | $\{ x \in \mathbb{R}^n : \sum_{i=1}^{n} x_i = 1 \}$ | $\sum_{i=1}^{n} x_i \log(x_i)$ | $\co\{ \frac{\exp(\|y\|_2^2)}{\exp(y)} \}_i=1$ |
| Euclidean sphere $\mathbb{B}^n_\omega(w)$ | $\{ x \in \mathbb{R}^n : \|x - \omega\|^2 \leq p \}$ | $-\sqrt{p^2 - \|x - \omega\|^2}$ | $py/\sqrt{1 + \|y\|^2} - w$ |

Algorithm 1: MDBD for agent $i \in V$.

Initialization:

1) For $i \in V$, the local set constraint $\Omega_i$ is closed and convex, and functions $f_i$, $g_i$, $A_i$, and $b_i$. Thus, agents need communication with neighbors through the network $G$.

Remark 1: Clearly, (2) can be regarded as a generalization for both distributed optimal consensus problems [12], [24], [36] and distributed resource allocation problems [13], [16]. Moreover, $g_i$ in the coupled constraints may not require to be affine, which is more general than the constraints in previous works [10], [27]. Also, the problem setting does not require strong or strict convexity for either cost functions $f_i$ or constraint functions $g_i$.

Algorithm 1: MDDB for agent $i \in V$.

Initialization:

1) For $i \in V$, the local set constraint $\Omega_i$ is closed and convex, and functions $f_i$, $g_i$, $A_i$, and $b_i$. Thus, agents need communication with neighbors through the network $G$.

Assumption 1:

1) For $i \in V$, the local set constraint $\Omega_i$ is closed and convex, and functions $f_i$, $g_i$, $A_i$, and $b_i$. Thus, agents need communication with neighbors through the network $G$.

2) There exists at least one $x \in \text{rint}(\Omega)$ such that $\sum_{i=1}^{N} g_i(x_i) < b_0$ and $\sum_{i=1}^{N} A_i x_i - b_i = 0$.

3) Graph $G$ is undirected and connected.

In order to solve problem (2) in a distributed manner, each agent involves its local Lagrange multipliers $\lambda_i \in \mathbb{R}^p_+$ and $\mu_i \in \mathbb{R}^q$ to handle inequality constraints and equality constraints individually, and exchanges this information with local neighbors through network $G$ to exchange consensus multipliers. Denote $\lambda = \text{col}\{\lambda_i\}_{i=1}^{N} \in \mathbb{R}^{Np}_+$, and $\mu = \text{col}\{\mu_i\}_{i=1}^{N} \in \mathbb{R}^{Nq}$. Take $\omega = \text{col}\{\omega_i\}_{i=1}^{N} \in \mathbb{R}^{Np}$, $\nu = \text{col}\{\nu_i\}_{i=1}^{N} \in \mathbb{R}^{Nq}$ as the collection of local auxiliary variables to deal with additional constraints $\lambda_i = \lambda_j$, $\mu_i = \mu_j$, or equivalently, $L_\lambda = 0_{N \times N}$ and $L_\mu = 0_{N \times N}$ with $L_\rho = L_N \otimes I_\rho$ and $L_\omega = L_N \otimes I_\omega$. Then, we can construct an augmented Lagrangian function $\tilde{L}_\omega: \Omega \times \mathbb{R}^{Np}_+ \times \mathbb{R}^{Nq}$ as follows:

\[
\tilde{L}_\omega(x, \lambda, \mu, \omega, \nu) = \sum_{i=1}^{N} f_i(x_i) + \sum_{i=1}^{N} \lambda_i^T (g_i(x_i) - \sum_{j=1}^{N} a_{ij} (\omega_j - \omega_i)) + \sum_{i=1}^{N} \mu_i^T (A_i x_i - b_i - \sum_{j=1}^{N} a_{ij} (\nu_j - \nu_i)).
\]

Moreover, let us denote

\[
z = \text{col}\{x, \lambda, \mu, \omega, \nu\}, \quad \Theta = \Omega \times \mathbb{R}^{Np}_+ \times \mathbb{R}^{Nq}.
\]

By Assumption 1, we can learn a closed duality gap of the Lagrange function above. Thus, $x^* \in \Omega$ is an optimal solution to problem (2) if and only if there exist auxiliary variables $(\lambda^*, \mu^*, \omega^*, \nu^*) \in \mathbb{R}^{Np}_+ \times \mathbb{R}^{Nq} \times \mathbb{R}^{Nq}$, such that $z^* = \text{col}\{x^*, \lambda^*, \mu^*, \omega^*, \nu^*\}$ is a saddle point of $\tilde{L}_\omega$, i.e., for arbitrary $z \in \Theta$,

\[
\tilde{L}_\omega(z^*, \lambda^*, \mu^*, \omega^*, \nu^*) \leq \tilde{L}_\omega(z^*, \lambda^*, \mu^*, \omega^*, \nu^*) \leq \tilde{L}_\omega(z, \lambda^*, \mu^*, \omega, \nu).
\]

Readers can refer to [16], [37] for more details.

IV. ALGORITHM

Based on the distributed primal-dual framework, we consider employing MD methods to solve the nonsmooth constrained (2). Actually, MD replaces the Euclidean regularization in (sub)gradient descent algorithms with Bregman divergence. In return, different generating functions of Bregman divergence may efficiently bring explicit solutions on different special feasible sets such as the simplex and the unit square. Readers can check Table I for more examples.

Under multiagent protocols, assign a generating function $\phi_i: \mathbb{R}^n \rightarrow \mathbb{R}$ of the Bregman divergence for each agent $i \in V$, which is differentiable and $\kappa$-strongly convex on $\Omega_i$. Such a selection qualification for generating function $\phi_i$ has also been widely discussed in the literature [18], [24], [28]. In order to develop a distributed MD-based algorithm suitable for the primal-dual framework, we employ the gradient $\nabla \phi_i(x_i)$ of generating functions as the Bregman damping. This idea plays an essential role in ensuring all variable trajectories’ boundedness [18], [24] and avoiding inaccurate solutions [28]. $a_{ij}$ is the $(i, j)$th entry of the adjacency matrix $A$ and the mirror map $\Pi_\Omega^\phi (x_i)$ is defined in (1). Then, we propose a distributed MD-based dynamics with Bregman damping (MDBD) in Algorithm 1 for solving (2).

To further clarify the design of MDBD, we provide details for the variables therein, including $y_i, x_i$, and other variables. First, to update the decision variable $x_i \in \Omega_i$, each agent employs the private gradient information to compute $y_i$ in the conjugate space. The term $\nabla \phi_i(x_i)$
is regarded as both a modified term to restrict the update direction of $y_i$ and a damping term to avoid $y_i$ going to infinity. Each agent then adopts mirror map $\Pi^\phi_{\Omega_i}(y_i)$ from conjugate spaces back to primal spaces to obtain $x_i$.

In fact, the entire dynamics flows of MDBD can be viewed from the MD perspective, not only for $x_i$ and $y_i$. Take $\frac{1}{2}\|\cdot\|^2$ as the generating function for variables $\lambda_i, \mu_i, \omega_i, \nu_i$ on their respective domains. Specifically, the update of local Lagrange multiplier $\lambda_i \in \mathbb{R}_+^p$ is similar to that of $x_i$, where $\gamma_i$ can be regarded as the conjugate variable of $\lambda_i$. Its associated Bregman damping is $\nabla \frac{1}{2}\|\lambda_i\|^2 = \lambda_i$ and the mirror map implies $\Pi^\phi_{\mathbb{R}_+^p}(\gamma_i) = [\gamma_i]^+$. Analogously, the domain of $\mu_i$ is the entire space $\mathbb{R}^q$, which indicates that the conjugate variable of $\mu_i$ is itself, that is, $\nabla \frac{1}{2}\|\mu_i\|^2 = \mu_i$, and $\Pi^\phi_{\mathbb{R}_+^q}(\mu_i) = \mu_i$. Thus, we can omit its mirror map, and the update of $\mu_i$ merely turns to $\mu_i = A_ix_i - b_i - \sum_{j=1}^N a_{ij}(\nu_i - \nu_j) + \mu_i - \mu_i = A_ix_i - b_i - \sum_{j=1}^N a_{ij}(\nu_i - \nu_j)$. The design ideas for $\omega_i \in \mathbb{R}^p$ and $\nu_i \in \mathbb{R}^q$ are the same as that for $\mu_i$, since their domains are also unconstrained. Hence, $\nabla \frac{1}{2}\|\omega_i\|^2 = \omega_i$, and $\nabla \frac{1}{2}\|\nu_i\|^2 = \nu_i$. Also, $\Pi^\phi_{\mathbb{R}_+^p}(\omega_i) = \omega_i$, and $\Pi^\phi_{\mathbb{R}_+^q}(\nu_i) = \nu_i$.

Additionally, in MDBD, information like $\partial f_i, \partial g_i, A_i$, and $b_i$ serves as private knowledge of each agent $i \in V$, and values like $\omega_i, \nu_i, \lambda_i$, and $\mu_i$ should be exchanged with neighbors through the network $G$. Meanwhile, generating function $\phi_i$ and the corresponding Bregman damping $\nabla \phi_i$ can be determined privately and individually, not necessarily identical. It follows from Lemma 2 that the existence of a Caratheodory solution to Algorithm 1 can be guaranteed.

Following the above-mentioned explanation, we further present the compact MD form of MDBD. Define

$$y = \text{col}\{y_i\}_{i=1}^N \in \mathbb{R}^{N_n}, \quad \gamma = \text{col}\{\gamma_i\}_{i=1}^N \in \mathbb{R}^{N_p},$$

define the collection of all conjugate variables by $s = \text{col}\{y, \gamma, \mu, \omega, \nu\}$. Accordingly, the collection of Bregman damping terms and mirror maps of MDBD are regarded as

$$\nabla \Phi(z) \triangleq \text{col}\{\nabla \phi_i(x_i)\}_{i=1}^N, \text{col}\{\lambda_i\}_{i=1}^N, \text{col}\{\mu_i\}_{i=1}^N, \text{col}\{\omega_i\}_{i=1}^N, \text{col}\{\nu_i\}_{i=1}^N, \text{col}\{\nu_i\}_{i=1}^N, \text{col}\{\nu_i\}_{i=1}^N, \text{col}\{\nu_i\}_{i=1}^N,$$

$$\Pi^\phi_{\Omega_i}(s) \triangleq \text{col}\{\Pi^\phi_{\Omega_i}(y_i)\}_{i=1}^N, \text{col}\{[\gamma_i]^+\}_{i=1}^N, \text{col}\{\mu_i\}_{i=1}^N, \text{col}\{\omega_i\}_{i=1}^N, \text{col}\{\nu_i\}_{i=1}^N.$$

Define the augmented gradient

$$F(z) = \begin{bmatrix}
\text{col}\{\partial f_i(x_i) + \partial g_i(x_i)^T \lambda_i + A_i^T \mu_i\}_{i=1}^N \\
\text{col}\{-g_i(x_i)\}_{i=1}^N + L_i \omega_i \\
\text{col}\{-A_ix_i + b_i\}_{i=1}^N + L_i \nu_i \\
-L_i \lambda_i \\
-L_i \mu_i
\end{bmatrix}.$$

Then, Algorithm 1 can be equivalently written as

$$\begin{cases}
\dot{s} &= -\eta + \nabla \Phi(z) - s, \\
z &= \Pi^\phi_{\Omega_i}(s),
\end{cases}$$

where $\eta \in F(z)$. To sum up, the update flow of conjugate $s$ uses the gradient information and the Bregman damping $\nabla \Phi(z)$ to ensure the boundedness and convergence. The updated flow of $z$ uses mirror map $\Pi^\phi_{\Omega_i}(s)$ to implement the mapping from conjugate spaces back to the primal spaces. Moreover, it follows from [16, 37] that $z^*$ is a saddle point of $\bar{L}$ if and only if $-F(z^*) \in \mathcal{N}_{\Theta}(z^*)$.

In fact, if all the involved generating functions satisfy $\phi(\cdot) = \frac{1}{2}\|\cdot\|^2$, then it follows from Table 1 that the mirror map turns into the classical Euclidean regularization with projection operations, i.e., $\Pi^\phi_{\mathbb{R}_+^p}(s) = P_{\mathbb{R}_+^p}(s)$, and $\Pi^\phi_{\mathbb{R}_+^q}(s) = P_{\mathbb{R}_+^q}(s) = [s]^+$. Thus, (6) can be rewritten as

$$\begin{cases}
\dot{s} &= -\eta + z - s, \\
z &= P_{\mathbb{R}_+^q}(s),
\end{cases}$$

which is actually a widely-investigated dynamics, such as the proportional-integral-protocol in [8] and projected output feedback in [16]. In this view, we learn that MDBD not only generalizes the conventional distributed projection-based design for constrained optimization, but also establishes a bridge between the distributed primal-dual and MD methods. Clearly, $z$ in the first ODE in (7) does not derive from the variable $z$ itself, but actually from the gradient of the quadratic function $\frac{1}{2}\|z\|^2$ instead.

**Remark 2:** If we further simplify the problem with neither coupled constraints nor local constraints [18, 24], then the normal cone $\mathcal{N}_{\Theta}(z^*)$ consists of a unique vector. The unique conjugate relation holds on the entire space such that $z = \nabla \Phi(s) = \Pi^\phi_{\Omega_i}(s) \iff \nabla \Phi(z) = s$. Hence, in these unconstrained cases, (6) can be further simplified as follows without the damping term

$$\begin{cases}
\dot{s} &= -\eta, \\
z &= \Pi^\phi_{\Omega_i}(s).
\end{cases}$$

Therefore, it is important to employ the Bregman damping term in MDBD to handle the distributed optimization with both local constraints and coupled constraints. Actually, the Bregman damping $\nabla \Phi(z)$ in (6) is fundamental to the trajectory of variable $s$ avoid going to infinity [18], or converging to inexact solutions [28].

V. CONVERGENCE

In this section, we investigate the convergence of MDBD. Although Bregman damping improves the convergence of MDBD, it brings challenges for the convergence analysis. The following lemma shows a relationship between MDBD and the saddle points of the Lagrangian function $\bar{L}$.

**Lemma 3:** Under Assumption 1, $z^*$ is a saddle point of Lagrangian function $\bar{L}$ in (3) if and only if there exists $s^* \in -F(z^*) + \nabla \Phi(z^*)$ such that $z^* = \Pi^\phi_{\Omega_i}(s^*)$.

**Proof:** For $\tilde{z} \in \Pi^\phi_{\Omega_i}(s^*)$, its first-order condition is

$$-F(z^*) + \nabla \Phi(z^*) - \nabla \Phi(\tilde{z}) \in \mathcal{N}_{\Theta}(\tilde{z}).$$

We first show the sufficiency. Given $z^*$, suppose that there exists $s^* \in -F(z^*) + \nabla \Phi(z^*)$ such that $z^* = \Pi^\phi_{\Omega_i}(s^*)$. Thus, (9) holds with $\tilde{z} = z^*$, and $-F(z^*) \in \mathcal{N}_{\Theta}(z^*)$, which means that $z^*$ is a saddle point of $\bar{L}$.

Second, we show the necessity. Suppose $-F(z^*) \in \mathcal{N}_{\Theta}(z^*)$ and take $s^* \in -F(z^*) + \nabla \Phi(z^*)$. Recall that (9) holds with $\tilde{z} = z^*$, which implies that $z^*$ is a solution to $\Pi^\phi_{\Omega_i}(s^*)$. Since $\phi(\cdot) = \frac{1}{2}\|\cdot\|^2$ are strongly convex, the solution to $\Pi^\phi_{\Omega_i}(s^*)$ is unique. Therefore, $z^* = \Pi^\phi_{\Omega_i}(s^*)$.

The following theorem shows the convergence of MDBD.

**Theorem 1:** Under Assumption 1,

1) the trajectory $(s(t), z(t))$ of (6) is bounded;
2) $x(t)$ converges to an optimal solution to problem (2).

**Proof:** (i) First, we show that the output $z(t)$ is bounded. By Lemma 3, take $z^*$ as a saddle point of $\bar{L}$ and thus, there exists $s^* \in -F(z^*) + \nabla \Phi(z^*)$ such that $z^* = \Pi^\phi_{\Omega_i}(s^*)$. Take $\phi^*$ as the convex conjugate of $\phi_i$, and construct a Lyapunov candidate function
\[ V_i = \sum_{i=1}^{N} D_\gamma(y_i, y_i^*) + \frac{1}{2}\left(\|\gamma - [\gamma^*] + \| - \left[\| - [\gamma^*]\right]\right)
+ \frac{1}{2}\|\mu - [\mu^*]\|^2 + \frac{1}{2}\|\omega - [\omega^*]\|^2 + \frac{1}{2}\|\nu - [\nu^*]\|^2. \] (10)

Since \( x_i = \Pi_{\mathcal{O}_i}(y_i) \), it follows from Lemma 1 that
\[ \phi_i^*(y_i) = x_i^T y_i - \phi_i(x_i), \quad \phi_i^*(y_i) = x_i^T y_i^* - \phi_i(x_i^*). \]

The Bregman divergence becomes
\[ D_\gamma(y_i, y_i^*) = \phi_i^*(y_i) - \phi_i^*(y_i^*) - \nabla \phi_i^*(y_i)^T (y_i - y_i^*) = \phi_i^*(x_i) - \phi_i(x_i) - (x_i^* - x_i)^T y_i. \]

Since \( \phi_i(x) \) is \( \kappa \)-strongly convex for \( i \in \mathcal{V} \), we have
\[ \sum_{i=1}^{N} D_\gamma(y_i, y_i^*) \geq \frac{\kappa}{2}\|x - x^*\|^2 + \sum_{i=1}^{N} (x_i - x_i^*)^T (\nabla \phi_i(x_i) - y_i). \]

In fact, \( \nabla \phi_i(y_i) = \arg\min_{y_i \in \mathcal{O}_i} \{-x^T y_i + \phi_i(x)\} \) and
\[ 0 \leq (\nabla \phi_i(\nabla \phi_i(y_i)) - y_i)^T (\nabla \phi_i(y_i^*) - \nabla \phi_i(y_i)) \]
\[ = (\nabla \phi_i(x_i) - y_i)^T (x_i^* - x_i). \]

Thus, \[ \sum_{i=1}^{N} D_\gamma(y_i, y_i^*) \geq \frac{\kappa}{2}\|x - x^*\|^2. \] In addition, \( \|\lambda - [\lambda^*]\|^2 \leq \|\gamma - [\gamma^*]\|^2 - \|\gamma - [\gamma^*]\|^2 \).

Therefore, \[ V_i(s(t)) \geq \frac{\kappa}{2}\|x - x^*\|^2 + \|\lambda - [\lambda^*]\|^2 + \|\mu - [\mu^*]\|^2 + \|\omega - [\omega^*]\|^2 + \|\nu - [\nu^*]\|^2, \]
where \( \tau = \min\{\kappa, 1\} \). This means \[ V_i(s(t)) \geq \frac{\kappa}{2}\|x - x^*\|^2, \] that is, \( V_i \) is radially unbounded in \( z \).

Clearly, the function \( V_i \) along (20) satisfies
\[ \bar{\mathcal{E}}_x V_i = \left\{ \beta \in \mathbb{R} : \beta \geq \sum_{i=1}^{N} (\nabla \phi_i(y_i) - \nabla \phi_i(y_i^*))^T y_i
+ (\lambda - [\lambda^*])^T \gamma + (\mu - [\mu^*])^T \mu + (\omega - [\omega^*])^T \omega + (\nu - [\nu^*])^T \nu
= \left\{ \beta \in \mathbb{R} : \beta \geq (z - z^*)^T \left(-\eta + \nabla \phi_i(z) - s\right), \eta \in \mathcal{F}(z) \right\}. \]

With the saddle point \( z^* \), \( -(z - z^*)^T \eta \leq 0 \) since
\[ (z - z^*)^T \eta \geq \bar{\mathcal{E}}_x(x^*, \lambda^*, \mu^*, \omega^*, \nu^*) - \bar{\mathcal{E}}(x, \lambda^*, \mu^*, \omega^*, \nu^*). \] (11)

Furthermore, \( (\lambda - [\lambda^*])^T (\lambda - \gamma) = ([\lambda^*] + \lambda)^T (\lambda - \gamma) - \lambda^T \gamma \leq 0, \)
Hence, \( s(t) \) is bounded. Take another Lyapunov candidate function as \( V_2 = \frac{1}{2}\|y\|^2 \), which is radially unbounded in \( y \).

Along the trajectories of Algorithm 1, the derivative of \( V_2 \) satisfies
\[ \bar{\mathcal{E}}_x V_2 = \left\{ \zeta \in \mathbb{R} : \zeta = \sum_{i=1}^{N} y_i^T \left(-\partial f_i(x_i) - \partial g_i(x_i)^T \lambda_i \right.ight.
- A_i^T \mu_i + \nabla \phi_i(x_i)) - \|y_i\|^2 \}. \]

It is clear that, for a positive constant \( m_1, \zeta \leq -\|y\|^2 + m_1\|y\| = -2V_2 + m_1\sqrt{2V_2} \), since \( x, \lambda \), and \( \mu \) have been proved to be bounded. On this basis, it can be easily verified that \( V_2 \) is bounded, so is \( y \).

Analogously, take a Lyapunov candidate function as \( V_3 = \frac{1}{2}\|\gamma\|^2 \), which is radially unbounded in \( \gamma \).

Along the trajectories of Algorithm 1, the derivative of \( V_3 \) satisfies
\[ \sigma = \sum_{i=1}^{N} \gamma_i^T \left( g_i(x_i) - \sum_{j=1}^{N} a_{ij}(\omega_i - \omega_j) + \lambda_i - \gamma_i \right) - \|\gamma_i\|^2. \]

It is clear that \( \sigma \leq -\|\gamma\|^2 + m_2\|\gamma\| = -2V_3 + m_2\sqrt{2V_3} \) for a positive constant \( m_2 \), since \( x, \lambda \), and \( \omega \) have been proved to be bounded. On this basis, it can be easily verified that \( V_3 \) is bounded, so is \( \gamma \). Hence, \( s(t) \) is bounded.
where $\rho_i = -\partial f_i(x_i) - \partial g_i(x_i)T\lambda_i$. Equivalently, (13) can be expressed in a compact form

$$z^{k+1} = \Pi_\Theta^k(z^k - \alpha_k \eta^k),$$

where $\eta^k \in F(z^k)$ with the augmented gradient $F$ in (5). Consider the weighted averaged iterates in course of $k$ iterates as $x^k = \left(\sum_{j=1}^k \alpha_j\right)^{-1}\sum_{j=1}^k \alpha_j x^j$, and similarly define $\hat{\lambda}^k$, $\hat{\mu}^k$, $\hat{\omega}^k$, and $\hat{\nu}^k$. Then, we show the convergence rate of (13), as below.

**Theorem 3:** Under Assumption 1, if there exists a positive constant $D$ such that $\|\eta^k\| \leq D$, then (13) converges at a $O(1/k)$ rate with the step size $\alpha_k = D^{-1} \sqrt{\tau(\|z^k - z^*\|^2 + 2\tau^2 \|z^k - z^*\|^2)}$. Specifically, with $\tau = \min\{\kappa, 1\}$,

$$\mathcal{L}(x^k, \lambda^k, \mu^k, \omega^k, \nu^k) - \mathcal{L}(x^*, \lambda^*, \mu^*, \omega^*, \nu^*) \leq \sqrt{\frac{1}{k} \mathcal{I}^2}.$$

**Proof:** Take the collection of the Bregman divergence as

$$\Delta(z, z^*) = \sum_{i=1}^N D\phi_i(y_i, y_i') + \frac{1}{2} \left(\|\gamma - \gamma^+\|^2 - \|\gamma - \gamma^+\|^2\right) + \frac{1}{2} \|\mu - \mu^+\|^2 + \frac{1}{2} \|\omega - \omega^+\|^2 + \frac{1}{2} \|\nu - \nu^+\|^2.$$ (15)

Since $\phi_i(\cdot)$ is $\kappa$-strongly convex for $i \in V$, we have $\sum_{i=1}^N D\phi_i(y_i, y_i') \geq \frac{\kappa}{2}\|x^k - x^*\|^2$. Also, $\frac{1}{2}\left(\|\gamma - \gamma^+\|^2 - \|\gamma - \gamma^+\|^2\right) \geq \frac{\kappa}{2}\|\lambda - \lambda^\ast\|^2$. Therefore, with $\tau = \min\{\kappa, 1\}$,

$$\Delta(z, z^*) \geq \frac{\tau}{2}\|z - z^*\|^2.$$ (16)

By additionally employing Fenchel’s inequality [31],

$$\alpha_k \left(\|z - z^k\|^2 - \|z^* - z^k\|^2\right) \leq \frac{1}{2\alpha_k} \alpha_k \|\eta^k\|^2,$$ (17)

since the conjugate of a $\ell_2$ norm is a $\ell_2$ norm. Hence, we can make further scaling so that

$$\Delta(z^*, z^k) \leq \Delta(z^*, z^k) - \alpha_k \left(\|z^k - z^\ast\|^2 - \alpha_k \|\eta^k\|^2\right).$$

Moreover, by the property for saddle point $z^*$, we have $-\eta^k T \eta^k \leq 0$ similar to (11). By plugging this into (16) and rearranging the terms therein, we obtain

$$\alpha_k \left(\frac{1}{2} \|z - z^k\|^2\right) \leq \frac{1}{2\alpha_k} \|z^* - z^k\|^2 + \frac{1}{2\alpha_k} \alpha_k \|\eta^k\|^2.$$ (18)

**VII. EXPERIMENT**

In this section, we examine the correctness and effectiveness of Algorithm 1 on the classical simplex-constrained problems (see, e.g., [23], [28]), where the local constraint set is an $n$-simplex, e.g., $\Omega_i \{x_i \in \mathbb{R}_+^n: \sum_{t=1}^n x_{i,t} = 1\}, \forall i \in V$. First, we consider the following nonsmooth optimization problem with $N = 10$ and $n = 4:

$$\min_{x \in \Omega} \sum_{i=1}^N \|W_i x_i - d_i\|^2 + c_i \|x_i\|_1,$$

s.t. \[\sum_{i=1}^N g_i(x_i) \leq 0, \quad \sum_{i=1}^N a_i x_i - b_i = 0,\] (19)

where $W_i$ is a positive semidefinite matrix, $d_i \in \mathbb{R}^4$, and $c_i > 0$. The coupled inequality constraint is $g_i(x_i) = \|x_i\|^2 + c_i \|x_i\|_1 - \frac{\alpha_k}{2\alpha_k}$, while $a_i \in \mathbb{R}^{2 \times 4}$ and $b_i \in \mathbb{R}^2$ are random matrices ensuring the Slater’s constraint condition. Here, $W_i, d_i, g_i, A_i$, and $b_i$ are private to agent $i$, and all agents communicate through an undirected cycle network $G$; $1 \equiv 2 \equiv \cdots \equiv 10 \equiv 1$. To implement the MD method, we employ the negative entropy function as the generating function on $\Omega_i$ in Algorithm 1. In Fig. 1, we show the trajectories of one dimension of each $x_i$ and $y_i$, respectively. Clearly, the trajectories of both $x_i$ and $y_i$ in MDBD are bounded, while the boundedness of $y_i$ may not be guaranteed in [18] and [24].

Next, we show the effectiveness of MDBD by comparisons. As investigated in [18] and [20], when the generating function satisfies $\phi_i(x_i) = \sum_{i=1}^n x_{i,t} \log(x_{i,t})$ on the unit simplex, we have $\Pi_{\Theta}^0(y_i) = 1$.

![Fig. 1. Trajectories of all agents’ variables. (a) Trajectories of $x_i$. (b) Trajectories of $y_i$.](image-url)
TABLE II
REAL RUNNING TIME (SEC) IN DIFFERENT DIMENSIONS

|                | \( n = 4 \) | \( n = 64 \) | \( n = 256 \) | \( n = 1024 \) | \( n = 4096 \) | \( n = 10^5 \) | \( n = 10^6 \) |
|----------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| MDBD           | 0.47      | 2.42      | 6.76      | 12.98     | 27.99     | 146.62    | 466.60    |
| PIP-YANG       | 2.51      | 19.63     | 48.51     | 195.67    | 892.74    | > 3000    | > 5000    |
| POF-ZENG       | 3.92      | 21.78     | 39.73     | 207.03    | 1136.85   | > 3000    | > 5000    |

![Fig. 2. Optimal errors in different dimensions: \( n = 4, 16, 64, 256 \). (a) \( n = 4 \). (b) \( n = 16 \). (c) \( n = 64 \). (d) \( n = 256 \).](image)

In this circumstance, the MD-based method works better than projection-based algorithms, because the negative entropy allows the conjugate function to have a closed-form solution and is especially efficient in high-dimensional variables.

To this end, we investigate different dimensions of decision variable \( x_i \) and compare MDBD with two distributed continuous-time projection-based algorithms—the proportional-integral protocol (PIP-Yang) in [8] and the projected output feedback (POF-Zeng) in [16], still for the cost functions and the coupled constraints given in (18). In Fig. 2, the \( x \)-axis is for the real running time of the GPU, while the \( y \)-axis is for the optimal error \( \| x - x^* \| \). As the dimension increases, the real running time of the two projection-based dynamics is obviously longer than that of MDBD, because computing a mirror map is much faster than calculating a projection on high-dimensional constraint sets via solving a general quadratic optimization problem.

Furthermore, Table II lists the real running time with different problem dimensions. As the dimension increases, finding the projection points in large-scale circumstances becomes more and more time-consuming. But remarkably, our MDBD still maintains good performance.

VIII. CONCLUSION

In this article, we investigated a distributed nonsmooth optimization with both local constraints and coupled constraints. To efficiently solve the problem, we proposed a continuous-time MD-based algorithm under a primal-dual framework by introducing the Bregman damping. By virtue of nonsmooth techniques and conjugate functions, we provided the convergence and its \( O(1/t) \) convergence rate with respect to the duality gap in the augmented Lagrangian function. Meanwhile, by taking quadratic generating functions in the Bregman damping, we revealed that our MD-based approach could generalize the classic distributed Euclidian-projected dynamics, and established a connection between MD methods and the distributed primal-dual framework. For practical implementation, we further provided a discrete version of the proposed dynamics, and obtained an \( O(1/\sqrt{T}) \) convergence rate.

REFERENCES

[1] T. Yang et al., “A survey of distributed optimization,” Annu. Rev. Control, vol. 47, pp. 278–305, 2019.
[2] M. Zhu and S. Martínez, “On distributed convex optimization under inequality and equality constraints,” IEEE Trans. Autom. Control, vol. 57, no. 1, pp. 151–164, Jan. 2012.
[3] D. Yuan, D. W. Ho, and S. Xu, “Regularized primal-dual subgradient method for distributed constrained optimization,” IEEE Trans. Cybern., vol. 46, no. 9, pp. 2109–2118, Sep. 2016.
[4] A. Cherukuri and J. Cortés, “Initialization-free distributed coordination for economic dispatch under varying loads and generator commitment,” Automatica, vol. 74, pp. 183–193, 2016.
[5] J.-M. Xu and Y. C. Soh, “A distributed simultaneous perturbation approach for large-scale dynamic optimization problems,” Automatica, vol. 72, pp. 194–204, 2016.
[6] K. You, R. Tempo, and P. Xie, “Distributed algorithms for robust convex optimization via the scenario approach,” IEEE Trans. Autom. Control, vol. 64, no. 3, pp. 880–895, Mar. 2019.
[7] K. Lu, G. Jing, and L. Wang, “Online distributed optimization with strongly pseudoconvex-sum cost functions,” IEEE Trans. Autom. Control, vol. 65, no. 1, pp. 426–433, Jan. 2020.
[8] S. Yang, Q. Liu, and J. Wang, “A multi-agent system with a proportional-integral protocol for distributed constrained optimization,” IEEE Trans. Autom. Control, vol. 62, no. 7, pp. 3461–3467, Jul. 2017.
[9] Y. Zhu, W. Yu, G. Wen, G. Chen, and W. Ren, “Continuous-time distributed subgradient algorithm for convex optimization with general constraints,” IEEE Trans. Autom. Control, vol. 64, no. 4, pp. 1694–1701, Apr. 2019.
[10] S. Liang, X. Zeng, and Y. Hong, “Distributed nonsmooth optimization with coupled inequality constraints via modified Lagrangian function,” IEEE Trans. Autom. Control, vol. 63, no. 6, pp. 1753–1759, Jun. 2018.
[11] X. Li, L. Xie, and Y. Hong, “Distributed continuous-time nonsmooth convex optimization with coupled inequality constraints,” IEEE Trans. Control Netw. Syst., vol. 7, no. 1, pp. 74–84, Mar. 2020.
[12] W. Li, X. Zeng, S. Liang, and Y. Hong, “Exponentially convergent algorithm design for constrained distributed optimization via nonsmooth approach,” IEEE Trans. Autom. Control, vol. 67, no. 2, pp. 934–940, Feb. 2022.
[13] 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.
[14] T.-H. Chang, A. Nedić, and A. Scaglione, “Distributed constrained optimization by consensus-based primal-dual perturbation method,” IEEE Trans. Autom. Control, vol. 59, no. 6, pp. 1524–1538, Jun. 2014.
[15] S. Liang, X. Zeng, G. Chen, and Y. Hong, “Distributed sub-optimal resource allocation via a projected form of singular perturbation,” Automatica, vol. 121, 2020, Art. no. 109180.
[16] X. Zeng, P. Yi, Y. Hong, and L. Xie, “Distributed continuous-time algorithms for nonsmooth extended monotropic optimization problems,” SIAM J. Control Optim., vol. 56, no. 6, pp. 3973–3993, 2018.
[17] A. S. Nemirovskij and D. B. Yudin, Problem Complexity and Method Efficiency in Optimization. Hoboken, NJ, USA: Wiley, 1983.
[18] W. Krichene, A. Bayen, and P. Bartlett, “Accelerated mirror descent in continuous and discrete time,” Adv. Neural Inf. Process. Syst., vol. 28, pp. 2845–2853, 2015.
[19] J. Diakonikolas and L. Orecchia, “The approximate duality gap technique: A unified theory of first-order methods,” *SIAM J. Optim.*, vol. 29, no. 1, pp. 660–689, 2019.

[20] A. Ben-Tal, T. Margalit, and A. Nemirovski, “The ordered subsets mirror descent optimization method with applications to tomography,” *SIAM J. Optim.*, vol. 12, no. 1, pp. 79–108, 2001.

[21] A. Beck and M. Teboulle, “Mirror descent and nonlinear projected subgradient methods for convex optimization,” *Operations Res. Lett.*, vol. 31, no. 3, pp. 167–175, 2003.

[22] S. Shahrampour and A. Jadbabaie, “Distributed online optimization in dynamic environments using mirror descent,” *IEEE Trans. Autom. Control*, vol. 63, no. 3, pp. 714–725, Mar. 2018.

[23] D. Yuan, Y. Hong, D. W. Ho, and G. Jiang, “Optimal distributed stochastic mirror descent for strongly convex optimization,” *Automatica*, vol. 90, pp. 196–205, 2018.

[24] Y. Sun and S. Shahrampour, “Distributed mirror descent with integral feedback: Asymptotic convergence analysis of continuous-time dynamics,” *IEEE Control Syst. Lett.*, vol. 5, no. 5, pp. 1507–1512, Nov. 2021.

[25] Y. Wang, Z. Tu, and H. Qin, “Distributed stochastic mirror descent algorithm for resource allocation problem,” *Control Theory Technol.*, vol. 18, no. 4, pp. 339–347, 2020.

[26] Y. Yu and B. Açıkmeke, “RLC circuits-based distributed mirror descent method,” *IEEE Control Syst. Lett.*, vol. 4, no. 3, pp. 548–553, Jul. 2020.

[27] G. Chen, Y. Ming, Y. Hong, and P. Yi, “Distributed algorithm for ε-generalized Nash equilibria with uncertain coupled constraints,” *Automatica*, vol. 123, 2021, Art. no. 109313.

[28] B. Gao and L. Pavel, “Continuous-time discounted mirror-descent dynamics in monotone concave games,” *IEEE Trans. Autom. Control*, vol. 66, no. 11, pp. 5451–5458, Nov. 2021.

[29] T. T. Doan, S. Bose, D. H. Nguyen, and C. L. Beck, “Convergence of the iterates in mirror descent methods,” *IEEE Control Syst. Lett.*, vol. 3, no. 1, pp. 114–119, Jan. 2019.

[30] P. Yi and L. Pavel, “An operator splitting approach for distributed generalized Nash equilibria computation,” *Automatica*, vol. 102, pp. 111–121, 2019.

[31] A. Nedić and S. Lee, “On stochastic subgradient mirror-descent algorithm with weighted averaging,” *SIAM J. Optim.*, vol. 24, no. 1, pp. 84–107, 2014.

[32] D. Yuan, Y. Hong, D. W. Ho, and S. Xu, “Distributed mirror descent for online composite optimization,” *IEEE Trans. Autom. Control*, vol. 66, no. 2, pp. 714–729, Feb. 2021.

[33] Y. Sun and S. Shahrampour, “Linear convergence of distributed mirror descent with integral feedback for strongly convex problems,” in *Proc. 60th IEEE Conf. Decis. Control*, 2021, pp. 1683–1688.

[34] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[35] J. Cortés, “Discontinuous dynamical systems,” *IEEE Control Syst. Mag.*, vol. 28, no. 3, pp. 36–73, Jun. 2008.

[36] G. Shi and K. H. Johansson, “Randomized optimal consensus of multi-agent systems,” *Automatica*, vol. 48, no. 12, pp. 3018–3030, 2012.

[37] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Berlin, Germany: Springer, 2003.

[38] W. M. Haddad and V. Chellaboina, *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton, NJ, USA: Princeton Univ. Press, 2011.

[39] A. Nedić and A. Ozdaglar, “Subgradient methods for saddle-point problems,” *J. Optim. Theory Appl.*, vol. 142, no. 1, pp. 205–228, 2009.

[40] G. Chen and M. Teboulle, “Convergence analysis of a proximal-like minimization algorithm using Bregman functions,” *SIAM J. Optim.*, vol. 3, no. 3, pp. 538–543, 1993.