SIAM J. OPTIM.
Vol. 33, No. 2, pp. 652–683

UNDERSTANDING A CLASS OF DECENTRALIZED AND FEDERATED OPTIMIZATION ALGORITHMS: A MULTIRATE FEEDBACK CONTROL PERSPECTIVE*

XINWEI ZHANG†, MINGYI HONG†, AND NICOLA ELIA†

Abstract. Distributed algorithms have been playing an increasingly important role in many applications such as machine learning, signal processing, and control. Significant research efforts have been devoted to developing and analyzing new algorithms for various applications. In this work, we provide a fresh perspective to understand, analyze, and design distributed optimization algorithms. Through the lens of multirate feedback control, we show that a wide class of distributed algorithms, including popular decentralized/federated schemes, can be viewed as discretizing a certain continuous-time feedback control system, possibly with multiple sampling rates, such as decentralized gradient descent, gradient tracking, and federated averaging. This key observation not only allows us to develop a generic framework to analyze the convergence of the entire algorithm class, but, more importantly, it also leads to an interesting way of designing new distributed algorithms. We develop the theory behind our framework and provide examples to highlight how the framework can be used in practice.

Key words. distributed algorithms, convergence analysis, control perspective

MSC codes. 90C35, 90C30

DOI. 10.1137/22M1475648

1. Introduction.

Distributed computation has played an important role in popular applications such as machine learning, signal processing, and wireless communications, partly due to the dramatically increased size of the models and the datasets. In this paper, we consider a distributed system with N agents connected by a graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$, each optimizing a smooth and possibility nonconvex local function $f_i(x)$. The global optimization problem is formulated as [32]

$$\min_{\mathbf{x}\in\mathbb{R}^{Nd_x}} \quad f(\mathbf{x}) := \frac{1}{N}\sum_{i=1}^{N} f_i(x_i), \quad \text{s.t.} \quad x_i = x_j \; \forall\, (i,j) \in \mathbf{E}, \tag{1.1}$$

where $\mathbf{x} \in \mathbb{R}^{N\times d_x}$ stacks N local variables $\mathbf{x} := [x_1; \ldots; x_N]$; $x_i \in \mathbb{R}^{d_x}\, \forall\, i \in [N]$.

This problem has received much attention in recent years; see [3, 13] for a few recent surveys. Heterogeneous computational and communication resources in the distributed system create a number of different scenarios in distributed learning. Specifically, based on the application scenarios, we can roughly classify distributed optimization algorithms into those that solve Decentralized Optimization (DO) problems, that solve Federated Learning (FL) problems, and those that can achieve optimal resource utilization (OPT). Some of the related works are discussed below.

* Received by the editors February 3, 2022; accepted for publication (in revised form) November 2, 2022; published electronically June 6, 2023. Copyright is owned by SIAM to the extent not limited by these rights.
https://doi.org/10.1137/22M1475648
Funding: The first and second authors were partially supported by NSF grants CIF-1910385 and CMMI-1727757.
† ECE Department, University of Minnesota, Minneapolis, MN 55455 USA (zhan6234@umn.edu, mhong@umn.edu, nelia@umn.edu).

(a) When solving the DO problems, the agents are typically modeled as nodes on a communication graph, and the communication and computation resources are equally important. So the algorithms alternatingly perform communication and communication steps. For instance, the Decentralized Gradient Descent (DGD) algorithm [20, 34] extends gradient descent (GD) to the decentralized setting, where each agent performs one step of local gradient descent and local model average in each round. Other related algorithms such as the Decentralized Linearized ADMM (DLM) [17], the Decentralized Gradient Tracking (DGT) [35] and the in-Network succEssive conveX approximaTion (NEXT) [4] all utilize this kind of alternating update.

(b) The FL problems typically consider the setting that the clients are directly connected to a parameter-server, and that the communication at the server is the bottleneck of the system. The FL algorithms, such as the well-known FedAvg [1], perform multiple local updates before one communication step. However, when the data is *heterogeneous* among the agents, it is difficult for these algorithms to achieve convergence [10, 15]. Recent algorithms such as the FedProx [14], SCAFFOLD [9], and FedPD [36] have developed new techniques to improve upon FedAvg.

(c) There have been a number of recent algorithms which are designed to utilize the *minimum* computation and/or communication resources, while computing high-quality solutions. They typically perform multiple communication steps before one local update. For example, in [26] a multistep gossip protocol is used to achieve the optimal convergence rate in decentralized convex optimization; the xFilter [27] is designed for decentralized nonconvex problems, and it implements the Chebyshev filter on the communication graph, which requires multistep communication, and achieves the optimal dependency on the graph spectrum.

Despite the proliferation of distributed algorithms, there are a few concerns and challenges. First, for some hot applications, there are simply *too many algorithms* available, so much so that it becomes difficult to track all the technical details. Is it possible to establish some general guidelines to understand the relations between, and the fundamental principles of, those algorithms that provide similar functionalities? Second, much of the recent research on this topic appears to be *increasingly focused* on a specific setting (e.g., those mentioned in the previous paragraph). However, an algorithm developed for FL may have already been rigorously developed, analyzed, and tested for the DO setting; and vice versa. Since developing algorithms and performing analyses take significant time and effort, it is desirable to have some mechanisms in place to reduce the possibility of reinventing the wheel.

1.1. Contribution of this work. We argue that there is a strong demand for a framework of distributed optimization, which can help researchers and practitioners *understand* algorithm behaviors, *predict* algorithm performance, and *streamline* algorithm design. This paper intends to provide such a framework, for a substantial subclass of distributed algorithms, using tools from multirate feedback control systems. We will first show that a customized continuous-time feedback control system is well suited to model some key components (such as local computation, interagent communication) of distributed algorithms. We then show that when such a continuous-time system is discretized properly (i.e., different parts of the system adopt different sampling rates), it recovers a wide range of distributed optimization algorithms. Finally, we provide a generic convergence result that covers different feedback schemes and discretization patterns. The major benefits of our proposed framework are listed below.

(1) One can easily establish connections between a few subclasses of distributed algorithms that are developed for different settings. In some sense, they can be viewed as applying different discretization schemes to certain continuous-time control systems.
(2) It helps predict the algorithm performance. On the one hand, once the continuous-time control system and the desired discretization pattern are identified, and some sufficient conditions set forth by our framework are satisfied, one can readily obtain various system parameters as well as the convergence guarantees. On the other hand, if we found that an existing distributed algorithm performs poorly, it is likely because it does not fall into our framework (an example is provided to show such a case).
(3) It facilitates new algorithm design. Once the problem setting and the associated requirement are determined, one can start with selecting the desired controllers and feedback schemes for the continuous-time system, followed by finding the appropriate discretization patterns. The performance of the new algorithm can be again readily obtained from our framework (as discussed in the previous point).

Note that there are many existing works which analyze optimization algorithms using control theory, but they mainly focus on some very special classes of algorithms. For examples, [25] studies continuous-time gradient flow for convex problems; [29, 32] study continuous-time first-order convex optimization algorithms; [8, 12, 19] investigate the acceleration approaches including Nesterov and Heavy-ball momentum methods for centralized problems in discrete time and interpret them as discrete-time controllers; [19, 32] focus on the continuous-time system and ignore the impact of the discretization; [6, 30, 31] investigate the connection between continuous-time system and discretized gradient descent algorithm, but their approaches and analyses do not generalize to other federated/decentralized algorithms. Further, to the best of our knowledge, none of the above referred works provide insights about relationship between subclasses of distributed algorithms (e.g., between DO and FL).

Notations and assumptions. We introduce some useful assumptions and notations.

First, let $\otimes$ denote the Kronecker product. the incidence matrix A of a graph $\mathcal{G}$ is defined as follows: if edge $e(i,j) \in \mathbf{E}$ connects vertex i and j with $i > j$, then $A_{ei} = 1$, $A_{ej} = -1$, and $A_{ek} = 0 \, \forall k \neq i, j$. Let us use $\mathcal{N}_i \subset [N]$ to denote the neighbors for agent i. For a symmetric matrix X, let us use $\lambda(X)$ to denote its eigenvalues. Then we can write the constraint of (1.1) in a more compact form:

$$\min_{\mathbf{x} \in \mathbb{R}^{Nd_x}} \quad f(\mathbf{x}) := \frac{1}{N} \sum_{i=1}^{N} f_i(x_i), \quad \text{s.t.} \quad (A \otimes I) \cdot \mathbf{x} = 0.$$

For simplicity of notation, the Kronecker products are ignored in the subsequent discussion, e.g., we use $A\mathbf{x}$ in place of $(A \otimes I) \cdot \mathbf{x}$. Define the averaging matrix $R := \frac{\mathbb{1}\mathbb{1}^T}{N}$ and the average of x_i's as $\bar{\mathbf{x}} := \frac{\mathbb{1}^T}{N}\mathbf{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$. Note, we have $R^2 = R$. The consensus error can be written as $[x_1 - \bar{\mathbf{x}}, \ldots, x_N - \bar{\mathbf{x}}] = (I - R)\mathbf{x}$, and we have $\nabla f(\bar{\mathbf{x}}) = \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(\bar{\mathbf{x}})$. The stationary solution of (1.1) is defined as follows.

Definition 1.1 (first-order stationary point). *We define the first-order stationary solution and the ϵ-stationary solution respectively, as*

$$\sum_{i=1}^{N} \nabla f_i\left(\frac{1}{N}\sum_{i=1}^{N} x_i\right) = 0, \quad \mathbf{x} - \frac{\mathbb{1}\mathbb{1}^T}{N}\mathbf{x} = 0, \tag{1.2a}$$

$$\left\| \frac{1}{N}\sum_{i=1}^{N} \nabla f_i\left(\frac{1}{N}\sum_{i=1}^{N} x_i\right) \right\|^2 + \left\| \mathbf{x} - \frac{\mathbb{1}\mathbb{1}^T}{N}\mathbf{x} \right\|^2 \leq \epsilon. \tag{1.2b}$$

We refer to the left-hand side (LHS) of (1.2b) *as the stationarity gap of* (1.1).

We will make the following assumptions on problem (1.1) throughout this paper:

A 1 (graph connectivity). *The graph is fixed, and strongly connected at all time* $t \in [0, \infty)$*, i.e.,* 0 *is a simple eigenvalue of* $A^T A$*, with corresponding eigenvector* $\frac{\mathbb{1}}{\sqrt{N}}$.

This assumption can be extended to time-varying graphs (denoted as $A(t)$'s), as they can be treated as subsampling on a strongly connected graph $A = \bigcup_t A(t)$. However, to stay focused on the main point of this paper (e.g., build connection of different algorithms from the control perspective) and to reduce notation, we choose to consider the simple static graph $A(t) = A \, \forall \, t \in [0, \infty)$ in this work.

Since the agents are connected by a fixed communication graph, we can further define the averaging matrix of the communication graph as $W := I - A^T \mathrm{diag}(\mathbf{w})\mathrm{A}$, where $\mathbf{w}$ is a vector each of whose entries $\mathbf{w}[e(i,j)]$ is positive, and it corresponds to the weight of edge $e(i,j)$. It is easy to check that W has the following properties:

$$W = W^T, \quad \mathbb{1}^T W = \mathbb{1}^T, \quad W_{ij} \geq 0 \quad \forall e(i,j) \in \mathbf{E}. \tag{1.3}$$

A 2 (Lipschitz gradient). *The* f_i*'s have Lipschitz gradient with constant* L_f*:*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_f \|x - y\| \quad \forall \, x, y \in \mathbb{R}^{d_x}, \forall \; i \in [N].$$

A 3 (lower bounded functions). *Each* f_i *is lower bounded as*

$$f_i(x) \geq \underline{f}_i > -\infty \quad \forall x \in \mathbb{R}^{d_x}, \quad \forall i \in [N].$$

A 4 (coercive functions). *Each* f_i *approaches infinity as* $\|x\|$ *approaches infinity:*

$$f_i(x) \to \infty \text{ as } \|x\| \to \infty \quad \forall i \in [N].$$

A3 and A4 imply that there exists at least one globally optimal solution $\mathbf{x}^\star$ for problem (1.1). Let us denote the corresponding optimal objective as $f^\star := f(\mathbf{x}^\star)$.

2. Continuous-time system. We present a continuous-time feedback control system. We will provide a number of key properties of the controllers and the entire system to ensure that the system converges to the set of first-order stationary points with guaranteed speed. These properties will be instrumental when we subsequently analyze discretized version of the system (hence, various distributed algorithms).

2.1. System description. To optimize problem (1.1), our approach is to design a continuous-time feedback control system, such that the state variables belong to the set of stationary points of the system if and only if they correspond to a stationary solution of (1.1). Towards this end, define $\mathbf{x} \in \mathbb{R}^{Nd_x}$ as the main state variable of the system; introduce the *global consensus feedback loop* (GCFL) and *local computation feedback loop* (LCFL), where the former incorporates the dynamics from multiagent interactions and pushes $\mathbf{x}$ to consensus, while the latter helps stabilize the system and finds the stationary solution. Specifically, these loops are defined as below.

• (The GCFL)**.** Define an auxiliary state variable $\mathbf{v} := [v_1; \ldots; v_N] \in \mathbb{R}^{Nd_v}$, with $v_i \in \mathbb{R}^{d_v} \; \forall \; i$; define $\mathbf{y} := [\mathbf{x}; \mathbf{v}] \in \mathbb{R}^{N(d_x + d_v)}$; define a feedback controller $G_g(\cdot; A) : \mathbb{R}^{N(d_x+d_v)} \to \mathbb{R}^{N(d_x+d_v)}$. Then the GCFL uses $G_g(\cdot; A)$ to operate on $\mathbf{y}$ to ensure the agents remain coordinated, and their local control variables remain close to consensus.

• (The LCFL)**.** Define an auxiliary state variable $\mathbf{z} := [z_1; \ldots; z_N] \in \mathbb{R}^{Nd_z}$, with $z_i \in \mathbb{R}^{d_z} \; \forall \; i$; define a set of feedback controller $G_\ell(\cdot; f_i) : \mathbb{R}^{d_x+d_v+d_z} \to \mathbb{R}^{d_x+d_v+d_z}$, one

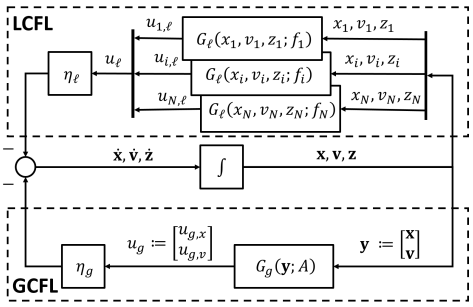


FIG. 1. *The proposed continuous-time double-feedback system for modeling the decentralized optimization problem* (1.1). *The system dynamics are given in* (2.5).

for each agent i. Then each agent will use LCFL to operate on its local state variables x_i, z_i and v_i, to ensure that its local system can be stabilized.

The overall system is described in Figure 1. The detailed description of properties of different controllers, as well as the notations used, will be given in the next sections.

To have a rough idea of how these loops can be mapped to a distributed algorithm, let us consider the Proportional-Integral (PI) distributed optimization algorithm [5], whose updates are

$$\dot{\mathbf{x}} = -k_G \nabla f(\mathbf{x}) - k_P \cdot (I - W) \cdot \mathbf{x} - k_P k_I \mathbf{v},$$
$$\dot{\mathbf{v}} = k_P k_I \cdot (I - W)\mathbf{x}.$$

The corresponding controllers are given by

$$G_g(\mathbf{x}, \mathbf{v}; A) := \begin{bmatrix} (I - W) \cdot \mathbf{x} + k_I \mathbf{v} \\ -k_I \cdot (I - W) \cdot \mathbf{x} \end{bmatrix}, \quad G_\ell(x_i, v_i, z_i; f_i) := \begin{bmatrix} \nabla f_i(x_i) \\ 0 \\ 0 \end{bmatrix},$$

with $\eta_\ell = k_G$ and $\eta_g = k_P$. Note that auxiliary state variable $\mathbf{z}$ has not been used in this algorithm.

Next, we describe in detail the properties of the two feedback loops.

2.2. Global consensus feedback loop. The GCFL performs interagent communication based on the incidence matrix A, and it controls the consensus of the global variable $\mathbf{y} := [\mathbf{x}; \mathbf{v}]$. Specifically, at time t, define the output of the controller as $u_g(t) = G_g(\mathbf{y}(t); A)$, which can be further decomposed into two outputs $u_g(t) := [u_{g,x}(t); u_{g,v}(t)]$, one to control the consensus of $\mathbf{x}$, and the other for $\mathbf{v}$. After being multiplied by the control gain $\eta_g(t) > 0$, the resulting signal will be combined with the output of the LCFL, and be fed back to local controllers.

We require that the global controller $G_g(\cdot; A)$ to have the following properties.

P 1 (control signal direction). *The output of the controller G_g aligns with the direction that reduces the consensus error, that is,*

$$\langle (I - R) \cdot \mathbf{y}, G_g(\mathbf{y}; A) \rangle \geq C_g \cdot \|(I - R) \cdot \mathbf{y}\|^2 \quad \forall\, \mathbf{y}$$

for some constant $C_g > 0$. Further, the controller G_g satisfies

$$\langle \mathbb{1}, G_g(\mathbf{y}; A) \rangle = 0 \quad \forall\, \mathbf{y}, \quad \text{which implies} \quad \langle \mathbb{1}, u_g(t) \rangle = 0 \quad \forall\, t.$$

P 2 (linear operator). *The controller G_g is a linear operator of $\mathbf{y}$, that is, we have $G_g(\mathbf{y}; A) = W_A\mathbf{y}$ for some matrix $W_A \in \mathbb{R}^{N(d_x+d_v)}$ parameterized by A, and its eigenvalues satisfy $|\lambda(W_A)| \in [0, 1]$.*

Combining P1 and P2, we have $\langle \mathbb{1}, W_A\rangle = 0$, which indicates $R \cdot W_A = 0$ and the eigenvectors of W_A are orthogonal to the ones of R. Further, we have

$$\begin{aligned}\|(I-R)\mathbf{y}\|^2 - \|G_g(\mathbf{y};A)\|^2 &= \mathbf{y}^T((I-R)^2 - W_A^2)\mathbf{y} \\ &= \mathbf{y}^T\left(I - 2R + R - W_A^2\right)\mathbf{y} = \mathbf{y}^T(I-(R+W_A^2))\mathbf{y}.\end{aligned}$$

Notice the eigenvectors of R and W_A are orthogonal and all eigenvalues are in $[0, 1]$, so we have matrix $I-(R+W_A^2) \succeq 0$. Thus $\mathbf{y}^T(I-(R+W_A^2))\mathbf{y} \geq 0$ and $\|(I-R)\mathbf{y}\|^2 \geq \|G_g(\mathbf{y};A)\|^2$. Therefore, we have

$$C_g^2 \left\|(I-R)\cdot\mathbf{y}\right\|^2 \leq \left\|G_g(\mathbf{y};A)\right\|^2 \leq \left\|(I-R)\cdot\mathbf{y}\right\|^2 \text{ and } R\cdot W_A = 0. \tag{2.1}$$

It is easy to check that both P1 and P2 hold in most of the existing consensus-based algorithms. For example, when the communication graph is strongly connected, we can choose $G_g(\mathbf{y}; A) = (I - W)\cdot\mathbf{y}$. It is easy to verify that $C_g = 1 - \lambda_2(W)$, where $\lambda_2(\cdot)$ denotes the eigenvalue withe the second largest magnitude [3, 34]. As another example, consider the accelerated averaging algorithms [7], where we have

$$G_g(\mathbf{y}, A) = \begin{bmatrix} I-(c+1)\cdot W & c\cdot I \\ -I & I \end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \mathbf{v}\end{bmatrix}, \quad \text{with } c := \frac{1-\sqrt{1-\lambda_2(W)}}{1+\sqrt{1-\lambda_2(W)^2}}.$$

In this case, one can verify that $C_g = 1 - \frac{\lambda_2(W)}{1+\sqrt{1-\lambda_2(W)^2}} \geq 1 - \lambda_2(W)$.

By using P1, we can follow the general analysis of averaging systems [21], and show that the GCFL will behave *as expected*, that is, if the system *only* performs GCFL and shuts off the LCFL, then the consensus can be achieved. More precisely, assuming that $\eta_\ell(t) = 0, \eta_g(t) = 1$, then under P1, the local state $\mathbf{y}$ converges to the average of the initial states linearly:

$$\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2 \leq \mathrm{e}^{-2C_g t}\left\|(I-R)\cdot\mathbf{y}(0)\right\|^2. \tag{2.2}$$

For completeness we include the derivation in the supplemental material section C.1.[1]

2.3. The local computation feedback loop. The LCFL optimizes the local function $f_i(\cdot)$'s for each agent. At time t, the ith local controller takes the local variables $x_i(t), v_i(t), z_i(t)$ as inputs and produces a local control signal. To describe the system, let us denote the output of the local controllers as $u_{i,\ell}(t) = G_\ell(x_i(t), v_i(t), z_i(t); f_i)\ \forall\ i \in [N]$; further decompose it into three parts:

$$u_{i,\ell}(t) := [u_{i,\ell,x}(t); u_{i,\ell,v}(t); u_{i,\ell,z}(t)].$$

Denote the concatenated local controller outputs as $u_{\ell,x}(t) := [u_{1,\ell,x}(t); \ldots; u_{N,\ell,x}(t)]$, and define $u_{\ell,v}(t), u_{\ell,z}(t)$ similarly. Note that we have assumed that all the agents use the same local controller $G_\ell(\cdot;\cdot)$, but they are parameterized by different f_i's. After multiplied by the control gain $\eta_\ell(t) > 0$, the resulting signal will be combined with the output of GCFL, and be fed back to the local controllers.

[1] Due to space limitation, less important results are relegated to the supplementary material; see [37].

The local controllers are designed to have the following properties:

P 3 (Lipschitz smoothness). *The controller is Lipschitz continuous, that is,*

$$\|G_\ell(x_i, v_i, z_i; f_i) - G_\ell(x'_i, v'_i, z'_i; f_i)\| \le L \|[x_i; v_i; z_i] - [x'_i; v'_i; z'_i]\|$$
$$\forall\, i \in [N],\, x_i, x'_i \in \mathbb{R}^{d_x},\, v_i, v'_i \in \mathbb{R}^{d_v},\, z_i, z'_i \in \mathbb{R}^{d_z}.$$

P 4 (control signal direction and size). *The local controllers are designed such that there exist initial values $x_i(t_0)$, $v_i(t_0)$, and $z_i(t_0)$ ensuring that the following holds:*

$$\langle \nabla f_i(x_i(t)), u_{i,\ell,x}(t) \rangle \ge \alpha(t) \cdot \|\nabla f_i(x_i(t))\|^2 \quad \forall\, t \ge t_0,$$

where $\alpha(t) > 0$ satisfies $\lim_{t\to\infty} \int_{t_0}^t \alpha(\tau) \mathrm{d}\tau \to \infty$.

Further, for any given x_i, v_i, z_i, the sizes of the control signals are upper bounded by those of the local gradients. That is, for some positive constants C_x, C_v, and C_z:

$$\|u_{i,\ell,x}\| \le C_x \|\nabla f_i(x_i)\|, \quad \|u_{i,\ell,v}\| \le C_v \|\nabla f_i(x_i)\|, \quad \|u_{i,\ell,z}\| \le C_z \|\nabla f_i(x_i)\|.$$

Let us comment on these properties. P3 is easy to verify for a given realization of the local controllers; P4 abstracts the convergence property of the local optimizer. This property implies that the update direction $-u_{i,\ell,x}(t)$ points to a direction that decreases the local objective. Note that it is postulated that x_i, v_i and z_i are initialized properly, because in some of the cases, improper initial values lead to nonconvergence of the local controllers (or, equivalently, the local algorithm). For example, for accelerated gradient descent method [2, 33], $z_i(t_0)$ should be initialized as $\nabla f_i(x_i(t_0))$.

By using P4, we can follow the general analysis of the gradient flow algorithms (e.g., [22]), and show that the LCFL will behave *as expected*, in the sense that the agents can properly optimize their local problems. More precisely, assume that $\eta_g(t) = 0, \eta_\ell(t) = 1$, that is, the system shuts off the GCFL. Assume that $G_\ell(\cdot;\cdot)$ satisfies P4, then each local system produces $x_i(t)$'s that satisfy

$$\min_\tau \|\nabla f_i(x_i(t+\tau))\|^2 \le \gamma(\tau) \cdot (f_i(x_i(t)) - \underline{f}_i), \tag{2.3}$$

where $\{\gamma(\tau)\}$ is a sequence of positive constants satisfying

$$\gamma(\tau) = \frac{1}{\int_0^t \alpha(\tau)\mathrm{d}\tau} \to 0 \quad \text{as } \tau \to \infty. \tag{2.4}$$

We include the proof of the above result in the online supplementary [37, sect. C.2].

To close this subsection, we note that the continuous-time system we have presented so far (cf. Figure 1) can be described using the following dynamics:

$$\begin{aligned} \dot{\mathbf{v}}(t) &= -\eta_g(t) \cdot u_{g,v}(t) - \eta_\ell(t) \cdot u_{\ell,v}(t), \\ \dot{\mathbf{x}}(t) &= -\eta_g(t) \cdot u_{g,x}(t) - \eta_\ell(t) \cdot u_{\ell,x}(t), \quad \dot{\mathbf{z}}(t) = -\eta_\ell(t) \cdot u_{\ell,z}(t). \end{aligned} \tag{2.5}$$

Additionally, throughout this paper, we will use u_g and G_g, u_ℓ, and G_ℓ interchangeably.

2.4. Convergence properties. We proceed to analyze the convergence of the continuous-time system. Towards this end, we define an energy-like function:

$$\mathcal{E}(t) := f(\bar{\mathbf{x}}(t)) - f^\star + \frac{1}{2} \|(I - R) \cdot \mathbf{y}(t)\|^2. \tag{2.6}$$

Note that $\mathcal{E}(t) \geq 0$ for all $t \geq 0$. It follows that its derivative can be expressed as

(2.7)
$$\dot{\mathcal{E}}(t) = -\left\langle \nabla f(\bar{\mathbf{x}}(t), \eta_\ell(t) \cdot \frac{\mathbb{1}^T}{N} u_{\ell,x}(t) \right\rangle + \langle (I-R)\cdot \mathbf{y}(t), \eta_g(t) u_g(t) + \eta_\ell(t) u_{\ell,y}(t) \rangle .$$

In the following, we study the convergence of $\mathcal{E}(t)$ and characterize the set of stationary points that the states satisfy $\dot{\mathcal{E}}(t) = 0$. We do not attempt to analyze the stronger property of *stability*, not only because such a kind of analysis can be challenging due to the nonconvexity of the local functions $f_i(\cdot)$'s, but more importantly, analyzing the convergence of $\mathcal{E}(t)$ is already sufficient for us to understand the convergence of the state variable $\mathbf{x}$ to the set of stationary solutions of problem (1.1), as we will show shortly.

To proceed, we require that the system satisfies the following property.

P 5 (energy function reduction). *The derivative of the energy function, $\dot{\mathcal{E}}(\cdot)$ as expressed in* (2.7)*, satisfies the following:*

(2.8)
$$\begin{aligned} &-\int_0^t \left(\left\langle \nabla f(\bar{\mathbf{x}}(\tau), \eta_\ell(\tau) \cdot \frac{\mathbb{1}^T}{N} u_{\ell,x}(\tau) \right\rangle + \langle (I-R)\cdot \mathbf{y}(\tau), \eta_g(\tau) u_g(\tau) + \eta_\ell(\tau) u_{\ell,y}(\tau) \rangle \right) \mathrm{d}\tau \\ &\leq -\int_0^t \left(\gamma_1(\tau) \cdot \left\| \nabla f(\bar{\mathbf{x}}(\tau)) \right\|^2 + \gamma_2(\tau) \cdot \| (I-R)\cdot \mathbf{y}(\tau) \|^2 \right) d\tau, \end{aligned}$$

where $\gamma_1(\tau), \gamma_2(\tau) > 0$ are some time-dependent coefficients.

P5 is a property about the entire continuous-time system. Although one could show that by using P1–P4, and by selecting $\eta_g(t)$ and $\eta_\ell(t)$ appropriately, this property can be satisfied with some *specific* $\gamma_1(\tau)$ and $\gamma_2(\tau)$ (cf. Corollary 2.2.), here we still list it as an independent property, because at this point we want to keep the choice of $\gamma_1(\tau)$, $\gamma_2(\tau)$ general; please see section 2.5 for a more detailed discussion.

Next, we will show that under P5, the continuous-time system will converge to the set of stationary points, and that $\mathbf{x}$ will converge to the set of stationary solutions of problem (1.1).

THEOREM 2.1. *Suppose* P5 *holds true. Then we have the following results:*
(1) *Further, suppose that* P1*,* P2, and P4 hold, then $\dot{\mathcal{E}} = 0$ *implies that the corresponding state variable $\mathbf{x}_s$ is bounded, and the following holds:*

$$\dot{\mathbf{x}}_s = 0, \quad \dot{\mathbf{v}}_s = 0, \quad \dot{\mathbf{z}}_s = 0, \quad u_g = 0, \quad u_\ell = 0. \tag{2.9}$$

Additionally, let us define the set $\mathbf{S}$ as below:

$$\mathbf{S} := \{ \mathbf{v}, \mathbf{z} \,|\; \eta_\ell u_{\ell,v} + \eta_g u_{g,v} = 0,\, u_{\ell,z} = 0,\, \eta_\ell u_{\ell,x} + \eta_g u_{g,x} = 0 \} .$$

If we assume that $\mathbf{S}$ is compact for any state variable $\mathbf{x}$ that satisfies the stationarity condition (1.2a)*, then the auxiliary state variables $\{\mathbf{v}(t)\}$ and $\{\mathbf{z}(t)\}$ are also bounded.*
(2) *The control system asymptotically converges to the set of stationary points, in that $\mathbf{x}(t)$ is bounded $\forall t \in [0, \infty)$, and $\dot{\mathcal{E}} \to 0$. Further, the stationary gap* (1.2b) *can be upper bounded by the following:*

(2.10)
$$\min_t \left\{ \| \nabla f(\bar{\mathbf{x}}(t)) \|^2 + \| (I-R)\cdot \mathbf{y}(t) \|^2 \right\} = \mathcal{O}\left(\max\left\{ \frac{1}{\int_0^T \gamma_1(\tau) \mathrm{d}\tau}, \frac{1}{\int_0^T \gamma_2(\tau) \mathrm{d}\tau} \right\} \right).$$

Proof. To show part (1), consider a set of states $\mathbf{x}_s, \mathbf{v}_s, \mathbf{z}_s$ in which $\dot{\mathcal{E}}(\mathbf{x}_s, \mathbf{v}_s) = 0$. P5 implies that $\nabla f(\bar{\mathbf{x}}_s) = 0$, and P4 implies $\|u_\ell\| \le (C_x + C_v + C_z)\|\nabla f(\bar{\mathbf{x}}_s)\| = 0$. Similarly, with P1 and P2 we have that $\langle u_g, (I-R)\mathbf{y}_s\rangle = 0$ and $\mathbb{1}^T u_g = 0$ so $u_g = 0$. Therefore $\dot{\mathbf{x}}_s = 0, \dot{\mathbf{v}}_s = 0, \dot{\mathbf{z}}_s = 0$. Combining $\nabla f(\bar{\mathbf{x}}_s) = 0$ and A4 implies that $\mathbf{x}_s$ is bounded. Note that the value of $\mathbf{v}(t), \mathbf{z}(t)$ may not be bounded, even if the system converges to a stationary solution. Using the compactness assumption on the set $\mathbf{S}$, it is easy to show that $\mathbf{v}(t), \mathbf{z}(t)$ are also bounded.

To show part (2), we can integrate $\dot{\mathcal{E}}(t)$ from $t = 0$ to T to obtain

$$\int_0^T \gamma_2(t) \left\|(I-R)\cdot \mathbf{y}(t)\right\|^2 \mathrm{d}t + \int_0^T \gamma_1(t) \left\|\nabla f(\bar{\mathbf{x}}(t))\right\|^2 \mathrm{d}t \le \mathcal{E}(0) - \mathcal{E}(T),$$

divide both sides by $\int_0^T \gamma_1(t)\mathrm{d}t$ or $\int_0^T \gamma_2(t)\mathrm{d}t$, we obtain (2.10). By P5 we know $\int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le 0\,\forall t$, but since $\mathcal{E}(t) \ge 0$, it follows that $\lim_{t\to\infty} \dot{\mathcal{E}}(t) = 0$. ∎

Note that without the compactness assumption, $\mathbf{v}$ and $\mathbf{z}$ can be unbounded. As an example, FedYogi uses AdaGrad for LCFL [24] where $\mathbf{v}(t)$ accumulates the norm of the gradients and does not satisfy the compactness assumption, so $\lim_{t\to\infty} \mathbf{v}(t) \to \infty$. Although such unboundedness does not affect the convergence of the main state variable in part (2), from the control perspective it is still desirable to have a sufficient condition to guarantee the boundedness of all state variables.

Part (2) of the above result indicates that if P5 is satisfied, not only will the system asymptotically converge to the set of stationary points, but more importantly, we can use $\{\gamma_1(t), \gamma_2(t)\}$ to characterize the rate in which the stationary gap of problem (1.1) shrinks. This result, although rather simple, will serve as the basis for our subsequent system discretization analysis.

2.5. Summary. So far, we have completed the setup of the continuous-time feedback control system, by specifying the state variables, the feedback loops, and by introducing a few desired properties of the local controllers and the entire system. In particular, we show that property P5 is instrumental in ensuring that the system converges to the set of stationary points. However, there are two key questions that remain answered:
(i) How to ensure property P5 for a given continuous-time feedback control system?
(ii) How to map the continuous-time system to a distributed optimization algorithm, and to transfer the convergence guarantees of the former to the latter?

There are two different ways to answer question (i). First, for a *generic* system that satisfies properties P1–P4, we can show that when the control gains $\eta_g(t), \eta_\ell(t)$ are selected appropriately, then P5 will be satisfied; see Corollary 2.2 below.

COROLLARY 2.2. *Suppose that* P1, P3, *and* P4 *are satisfied. By choosing* $\eta_g(t) = 1, \eta_\ell(t) = \mathcal{O}(1/\sqrt{T})$), P5 *holds true with* $\gamma_1(t) = \mathcal{O}(\eta_\ell(t))$, $\gamma_2(t) = \mathcal{O}(1)$ *Further,*

$$\min_t \left\{ \left\|\nabla f(\bar{\mathbf{x}}(t))\right\|^2 + \left\|(I-R)\cdot \mathbf{y}(t)\right\|^2 \right\} = \mathcal{O}\left(\frac{1}{\int_0^T \eta_\ell(\tau)\mathrm{d}\tau}\right) = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right).$$

The proof of the above result follows the steps used in analyzing distributed gradient flow algorithm [30]; see the online supplementary material [37, sect C.3].

The second answer to question (i) is that one can also verify P5 in a case-by-case manner for individual systems. In this way, it is possible that one can obtain larger gains $\eta_\ell(t), \eta_g(t)$, hence larger coefficients $\gamma_1(t)$ and $\gamma_2(t)$ to further improve the convergence rate estimate. In fact, verifying property P5 and computing the corresponding coefficients is a key step in our proposed analysis framework for distributed

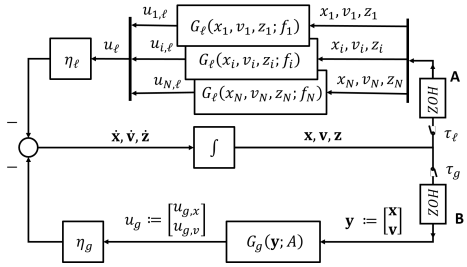


FIG. 2. *Discretized system using ZOH on both the GCFL and LCFL control loops with possibly different sampling times* τ_g, τ_ℓ. *The system dynamics are given in* (3.1)–(3.4).

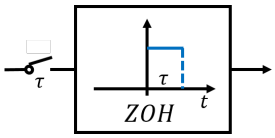


FIG. 3. *The discretization block that has a switch and a ZOH.*

algorithms. Shortly in section 4.1, we will provide an example to showcase how to verify that the continuous-time system which corresponds to the DGT algorithm satisfies P5 with $\gamma_1(t) = \mathcal{O}(1)$ and $\gamma_2(t) = \mathcal{O}(1)$, leading to a convergence rate of $\mathcal{O}(1/T)$.

On the other hand, the answer to question (ii) is more involved, so this question will be addressed in the main technical part of this work to be presented shortly. Generally speaking, one needs to discretize the continuous-time system properly to map the system to a particular distributed algorithm. Further, one needs to utilize all the properties P1–P5, and carefully select the discretization intervals to ensure that the resulting discretized systems perform appropriately.

3. System discretization. In this section, we discuss how to use system discretization to map the continuous-time system introduced in the previous section to distributed algorithms.

3.1. Modeling the discretization. Typically, a continuous-time system is discretized by using a switch that samples the input with sample time τ, followed by a zeroth-order hold (ZOH) that keeps the signal constant between the consecutive sampling instances [11]; see Figure 3.

Now let us use ZOH to discretize the continuous-time system depicted in Figure 1. We will place the ZOH before the variables enter the controllers, i.e., at points A and B in Figure 2. Note that, the original continuous-time system can be discretized in many different ways, by customizing the sampling rates for the discretization blocks. Each of these discretization scheme will correspond to a *multirate* control system, in which different parts of the system run on different sampling rates. To describe such kinds of multirate system, let us define the *sampling intervals* for the GCFL and LCFL as τ_g and τ_ℓ, respectively. Then we can consider the following five cases:

- *Case* I. $\tau_g > 0, \tau_\ell = 0$. The GCFL is discretized while the LCFL is not.
- *Case* II. $\tau_g = 0, \tau_\ell > 0$. The GCFL remains continuous while the LCFL is not.
- *Case* III. $\tau_g = \tau_\ell > 0$. The GCFL and LCFL are discretized with the same rate.

- *Case* IV. $\tau_g > \tau_\ell > 0$. Both the GCFL and LCFL are discretized, while the local computation loop is updated more frequently.
- *Case* V. $\tau_\ell > \tau_g > 0$, both GCFL and LCFL are discretized, while the global communication loop is updated more frequently.

We note that the systems in cases I and II are *sampled data* systems which has both continuous-time part and discretized part, while systems in cases IV, V are *multirate discrete-time* systems. Further, the entire system in case III operates on the same sampling rate. For simplicity, we refer to both the sampled data system and fully discretized system as the *discretized system* in the rest of this paper.

3.2. Distributed algorithms as multirate discretized systems. In this section, we make the connection between *subclasses* of distributed algorithms and different discretization patterns. For convenience, let t_k denote the times at which the inputs of the ZOHs get sampled by *both* the global and local controllers.

Case I. $(\tau_g > 0, \tau_\ell = 0)$. The system can be described as follows:

$$\begin{aligned} &\dot{\mathbf{v}}(t) = -\eta_g(t) \cdot u_{g,v}(t_k) - \eta_\ell(t) \cdot u_{\ell,v}(t), \\ &\dot{\mathbf{x}}(t) = -\eta_g(t) \cdot u_{g,x}(t_k) - \eta_\ell(t) \cdot u_{\ell,x}(t), \quad \dot{\mathbf{z}}(t) = -\eta_\ell(t) \cdot u_{\ell,z}(t). \end{aligned} \tag{3.1}$$

Due to the use of ZOH, during an interval $[t_k, t_k + \tau_g)$, the control signals $u_{g,v}$ and $u_{g,x}$ are fixed. By P4, it follows that the dynamic system finds a stationary point of the local problem satisfying $\dot{x}_i = 0 \ \forall \ i$, that is, $\eta_\ell(t) \cdot u_{\ell,x}(t) + \eta_g(t) \cdot u_{g,x}(t_k) = 0$. This is the stationary solution of the following perturbed problem for each agent:

$$\min_{x_i} \ \widetilde{f}_i(x_i) := f_i(x_i) + \frac{\eta_g(t)}{\eta_l(t)} \left\langle u_{i,g,x}(t_k), x_i \right\rangle. \tag{3.2}$$

Using (2.3), it follows that the above problem is optimized to satisfy

$$\min_{t \in [t_k, t_k + \tau_g]} \left\| \nabla \widetilde{f}_i(x_i(t)) \right\|^2 \leq \gamma(\tau_g) \cdot \left(\widetilde{f}_i(x_i(t_k)) - \widetilde{f}_i(x_i(t_k + \tau_g)) \right),$$

with $\gamma(\tau_g) = \frac{1}{\int_0^{\tau_g} \alpha(t) \mathrm{d}t}$. That is, we obtain a $\gamma(\tau_g)$-stationary solution for the local problem (3.2). This system has the same form as the distributed algorithms that require solving some local problems to a given accuracy, before any local communication steps take place; see, for example, FedProx [14], FedPD [36], and NEXT [4].

Case II $(\tau_g = 0, \tau_\ell > 0)$. The system can be described as follows:

$$\begin{aligned} &\dot{\mathbf{v}}(t) = -\eta_g(t) \cdot u_{g,v}(t) - \eta_\ell(t) \cdot u_{\ell,v}(t_k), \\ &\dot{\mathbf{x}}(t) = -\eta_g(t) \cdot u_{g,x}(t) - \eta_\ell(t) \cdot u_{\ell,x}(t_k), \quad \dot{\mathbf{z}}(t) = -\eta_\ell(t) \cdot u_{\ell,z}(t_k). \end{aligned} \tag{3.3}$$

During $[t_k, t_k + \tau_\ell)$ the control signals $u_{\ell,x}(t), u_{\ell,v}(t), u_{\ell,z}(t)$ are fixed. By P1, the system finds a solution $\dot{\mathbf{y}} = 0$, which implies that $-\eta_g(t) \cdot u_{g,x}(t) - \eta_\ell(t) \cdot u_{\ell,x}(t_k) = 0$. By (2.2), in $[t_k, t_k + \tau_\ell)$, the system optimizes the following network problem:

$$\min_{\mathbf{y}} g(\mathbf{y}) := \left\| (I - R) \cdot \mathbf{y} + (\eta_\ell(t)/\eta_g(t)) \cdot u_{\ell,y}(t_k) \right\|^2,$$

and obtain a solution that satisfies $\|\nabla g(\mathbf{y}(t_k + \tau_\ell))\|^2 \leq \mathrm{e}^{-2C_g \tau_\ell} g(\mathbf{y}(t_k))$. This system is related to those algorithms that achieve the optimal communication complexity [26, 27]. In these algorithms, it is often the case that some networked

TABLE 1
Summary of discretization settings, and the corresponding distributed algorithms.

Case	$\tau_\ell, \tau_{\mathbf{g}}$	Comm.	Comp.	Related algorithm
I	$\tau_g > 0, \tau_\ell = 0$	slow	continuous	NEXT [4], FedProx [14], NIDS [16]
II	$\tau_g = 0, \tau_\ell > 0$	continuous	slow	MSDA [26], xFilter [27],AGD [33]
III	$\tau_g = \tau_\ell > 0$	same rate		DGD [34], DGT [35]
IV	$\tau_g > \tau_\ell > 0$	slow	fast	Local GD [10], Scaffold [9]
V	$\tau_\ell > \tau_g > 0$	fast	slow	Same as case II

problems are solved (to sufficient accuracies) between two local optimization steps.

Case III ($\tau_g = \tau_\ell > 0$). The system is discretized with a single sampling interval. Once sampled at t_k, the controllers' inputs remain to be $\mathbf{x}(t_k), \mathbf{v}(t_k), \mathbf{z}(t_k)$ during the sampling interval, the output of the controllers are also kept constant $u_g(t) = u_g(t_k), u_\ell(t) = u_\ell(t_k) \forall t \in [t_k, t_k + \tau_g)$. So the system update can be written as

$$\begin{aligned} \mathbf{x}(t_{k+1}) &= \mathbf{x}(t_k) - \eta'_\ell(t_k) \cdot u_{\ell,x}(t_k) - \eta'_g(t_k) \cdot u_{g,x}(t_k), \\ \mathbf{v}(t_{k+1}) &= \mathbf{v}(t_k) - \eta'_\ell(t_k) \cdot u_{\ell,v}(t_k) - \eta'_g(t_k) \cdot u_{g,v}(t_k), \\ \mathbf{z}(t_{k+1}) &= \mathbf{z}(t_k) - \eta'_g(t_k) \cdot u_{\ell,z}(t_k), \end{aligned} \tag{3.4}$$

where $\eta'_\ell(t_k) = \int_{t_k}^{t_k+\tau_g} \eta_\ell(t)\mathrm{d}t$, $\eta'_g(t_k) = \int_{t_k}^{t_k+\tau_g} \eta_g(t)\mathrm{d}t$. The above updates are equivalent to many existing decentralized optimization algorithms, such as DGD, DLM, which perform one step local update, followed by one step of communication.

Case IV ($\tau_g > \tau_\ell > 0$). We assume that $\tau_g = Q \cdot \tau_\ell$, which means that each agent performs Q steps of local computation between every two communication steps. This update strategy is related to the class of (horizontal) federated learning algorithms [1].

Case V ($\tau_\ell > \tau_g > 0$). We assume that $\tau_\ell = K \cdot \tau_g$, and that the agents perform K steps of communication between two local computation steps. Although K can be arbitrary, in practice it is typically chosen large enough so that certain network problems are solved approximately; therefore, in practice this case is closely related to case II.

We summarize the above discussion in Table 1, and provide some example algorithms for each case. In section 4.1, we will specify the controllers for these algorithms so that we can precisely map them to a discretization setting. It is important to note that the connection identified here is useful in helping predict algorithm performance, as well as facilitating new algorithm design; see the related discussions in section 1.1, points (2) and (3). However, these benefits can be realized only if there is a systematic way of transferring the theoretical results from the continuous-time system to different discretization settings. This will be discussed in detail in the next subsection.

3.3. Convergence of discretized systems. Next, we leverage the convergence results of the continuous-time system to analyze distributed algorithms. The key challenge is to properly deal with the potential instability introduced by discretization. The proof of this subsection is relegated to Appendix A.1, A.2, and A.3.

Discretized communication ($\tau_g > 0, \tau_\ell = 0$**,** case I). Recall that the system dynamics are given in (3.1). Let us first show how the sampling error affects $\dot{\mathcal{E}}$.

LEMMA 3.1 ($\dot{\mathcal{E}}$ in case I). *Suppose the GCFL and LCFL satisfy* P1–P5*, and consider the discretized system with* $\tau_\ell = 0, \tau_g > 0$*. Then we have the following:*

$$\begin{aligned}\int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le \int_0^t &-\underbrace{(\gamma_1(\tau) - C_{11})}_{:=\hat{\gamma}_1(\tau)} \cdot \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \,\mathrm{d}\tau \\ &+ \int_0^t -\underbrace{\left(\frac{\gamma_2(\tau)}{2} - C_{11}\right)}_{:=\hat{\gamma}_2(\tau)} \cdot \|(I-R)\cdot \mathbf{y}(\tau)\|^2 \,\mathrm{d}\tau,\end{aligned} \tag{3.5}$$

where $C_{11} := \frac{q_{\max}^2}{2\gamma_2(\tau)}$ *and* $q_{\max} := \exp\left\{\sqrt{2}\tau_g \cdot \left(\sqrt{C_x^2 + C_v^2}\eta_\ell(t) \cdot \left(1 + \frac{L_f}{N}\right)^2\right)\right\} - 1.$

The lemma shows that discretizing the communication with sufficiently small τ_g leads to a small $q_{\max}$, which preserves the desired descent property.

Discretized computation ($\tau_\ell > 0, \tau_g = 0$**,** case II). Recall that the system dynamics can be expressed in (3.3). We have the following result.

LEMMA 3.2 ($\dot{\mathcal{E}}$ in case II). *Suppose the GCFL and LCFL satisfy* P1–P5*, and consider the discretized system with* $\tau_g = 0, \tau_\ell > 0$*. Then we have the following:*

$$\begin{aligned}\int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le \int_0^t &-\underbrace{\left(\frac{\gamma_1(\tau)}{2} - C_{21}\right)}_{:=\hat{\gamma}_1(\tau)} \cdot \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \,\mathrm{d}\tau \\ &+ \int_0^t -\underbrace{\left(\frac{\gamma_2(\tau)}{2} - C_{22}\right)}_{:=\hat{\gamma}_2(\tau)} \cdot \|(I-R)\cdot \mathbf{y}(\tau)\|^2 \,\mathrm{d}\tau,\end{aligned} \tag{3.6}$$

where we have defined

$$C_{21} := \frac{4L^2 C_f C_\ell^2 \eta_\ell^2(\tau)}{2(1-2L^2C_\ell^2)\cdot \min\{N\gamma_1(\tau), \gamma_2(\tau)\}}, C_{22} := \frac{L^2\eta_\ell^2(\tau)\cdot\left(\left(\frac{1-C_y}{C_y^2}\right) + 4L_f^2 C_f C_\ell^2\right)}{2(1-2L^2C_\ell^2)\cdot \min\{N\gamma_1(\tau), \gamma_2(\tau)\}},$$
$$C_f := C_x^2 + C_v^2 + C_z^2,\ C_y = \mathrm{e}^{-C_g\tau_\ell\eta_g(\tau)},\ C_\ell := \frac{\tau_\ell\eta_\ell(\tau)}{\min\{2C_g\eta_g(\tau), 1\}}.$$

Note that the requirements on $\hat{\gamma}_1(\tau) > 0, \hat{\gamma}_2(\tau) > 0$ result in the constraint on τ_ℓ, which will be discussed at the end of this section.

Two-sided discretization ($\tau_\ell > 0, \tau_g > 0$**,** cases III–V). We then analyze the more challenging cases where *both* the communication and the computation are discretized. Note that case III with $\tau_\ell = \tau_g > 0$ can be merged into case IV, with $Q = 1$.

LEMMA 3.3 ($\dot{\mathcal{E}}$ in cases III–IV). *Suppose the GCFL and LCFL satisfy properties* P1–P5*, and consider the discretized system with* $\tau_g = Q \cdot \tau_\ell$*. Then we have*

$$\begin{aligned}\int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le \int_0^t &-\underbrace{\left(\frac{\gamma_1(\tau)}{2} - C_{41}(\tau)\right)}_{:=\hat{\gamma}_1(\tau)} \cdot \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \,\mathrm{d}\tau \\ &+ \int_0^t -\underbrace{\left(\frac{\gamma_2(\tau)}{2} - C_{42}(\tau)\right)}_{:=\hat{\gamma}_2(\tau)} \cdot \|(I-R)\cdot \mathbf{y}(\tau)\|^2 \,\mathrm{d}\tau,\end{aligned} \tag{3.7}$$

where the constants $C_{41}(\tau)$ and $C_{42}(\tau)$ are defined as

$$C_{41} := \frac{L^2\eta_\ell^2(\tau)\cdot\left(C_{45}\cdot(1+L_f^2C_{47}+C_{45})+C_{46}L_f^2\right)}{2\min\{N\gamma_1(\tau),\gamma_2(\tau)\}}+\frac{C_g\eta_g^2(\tau)\cdot(C_{43}+L_f^2C_{47})}{2\gamma_2(\tau)},$$

$$C_{42} := \frac{L^2\eta_\ell^2(\tau)\cdot(C_{46}+C_{45}C_{47})}{2\min\{N\gamma_1(\tau),\gamma_2(\tau)\}}+\frac{C_g\eta_g^2(\tau)C_{47}}{2\gamma_2(\tau)},\quad C_{47} := Q^2C_{44}^2\cdot(C_x^2+C_v^2),$$

$$C_{43} := \frac{4\tau_g^2\eta_g^2(t)}{1-4\tau_g^2\eta_g^2(\tau)},\quad C_{44} := \frac{2\tau_\ell^2\tau_\ell^2(\tau)}{1-4\tau_g^2\eta_g^2(\tau)},$$

$$C_{45} := \frac{4\tau_\ell^2\eta_g^2(\tau)}{1-4L^2\tau_\ell^2\eta_\ell^2(\tau)},\quad C_{46} := \frac{8L^2C_f\tau_\ell^2\eta_\ell^2(\tau)}{1-4L^2\tau_\ell^2\eta_\ell^2(\tau)}.$$

Furthermore, we can check that when $\tau_g=0$ and $\tau_\ell=0$, then $C_{41}(\tau)$, $C_{42}(\tau)$ are both zero. Additionally, $\hat\gamma_1(\tau)>0,\hat\gamma_2(\tau)>0$ determine the upper bounds for τ_g,τ_ℓ, as well as the choice of the stepsizes of the discretized algorithms.

Finally, we note that for case V, a similar result with different $\hat\gamma_1(\tau),\hat\gamma_2(\tau)$ can be proved using the same technique as Lemmas 3.2 and 3.3. Since the utility of case V can be covered mostly by that of case II (cf. Table 1), and due to the space limitation, we will not discuss this case in detail here.

By using the above results, it is easy to obtain the following convergence characterization. The proof is straightforward and follows that of Theorem 2.1.

THEOREM 3.4 (convergence of the discretized systems). *Suppose the GCFL and LCFL satisfy properties* P1–P5*, and consider the discretized system with $\tau_\ell\ge 0,\tau_g\ge 0$. Then the convergence of the discretized system can be characterized as*

$$\min_t\left\{\|\nabla f(\bar{\mathbf{x}}(t))\|^2+\|(I-R)\cdot\mathbf{y}(t)\|^2\right\}=\mathcal{O}\left(\max\left\{\frac{1}{\int_0^T\hat\gamma_1(\tau)\mathrm{d}\tau},\frac{1}{\int_0^T\hat\gamma_2(\tau)\mathrm{d}\tau}\right\}\right),$$

where $\hat\gamma_1(\tau)>0$ and $\hat\gamma_2(\tau)>0$ depend on $\gamma_1(\tau),\gamma_2(\tau),N,C_g,L$ and $\eta_\ell,\eta_g,\tau_\ell,\tau_g,K,Q$, and their choices are specified in Lemmas 3.1*,* 3.2*, and* 3.3*.*

This result indicates that as long as $\hat\gamma_1(\tau)>0$ and $\hat\gamma_2(\tau)>0$, the discretized system preserves the convergence rate of the continuous-time system, but it slows down by a factor $\max\{\gamma_1(\tau)/\hat\gamma_1(\tau),\gamma_2(\tau)\hat\gamma_2(\tau)\}$. Further, the condition that $\hat\gamma_1(\tau)>0,\hat\gamma_2(\tau)>0$ give a way to decide the maximum sampling intervals and the choice of the hyperparameters (e.g., stepsize, the number of communication steps and local update steps K,Q) for different algorithms, as we explain below.

Let us consider case I first. By Lemma 3.1,

$$\min\{\gamma_2,2\gamma_1\}\ge\frac{q_{\max}^2}{\gamma_2},\ \text{with}\ q_{\max}=\mathrm{e}^{\sqrt{2}\tau_g\cdot\left(\sqrt{C_x^2+C_v^2}\eta_\ell(t)\cdot\left(1+\frac{L_f}{N}\right)^2\right)}-1.$$

It follows that $\tau_g\le\frac{\ln(\min\{\gamma_2(t),\sqrt{2\gamma_1(t)\cdot\gamma_2(t)}\}+1)}{\sqrt{2}\sqrt{C_x+C_v}\eta_\ell(t)\cdot\left(\frac{L_f}{N}+1\right)^2}$. Note that all the variables on the right-hand side (RHS) can be determined from the continuous-time system. This indicates that by having a convergent continuous-time system, the maximum sampling interval of the GCFL can be determined. Similarly, for case II, by Lemma 3.2, $\gamma_1(t)\ge 2C_{21}$, $\gamma_2(t)\ge 2C_{22}$, which implies:

$$\tau_\ell\le\min\left\{\frac{\tilde\gamma_1(t)}{\sqrt{2(\tilde\gamma_1^2(t)+4C_f)}L\eta_\ell^2(t)},\frac{\log\left(\frac{\tilde\gamma_2(t)+2L\eta_\ell(t)}{2L\eta_\ell(t)}\right)}{C_g\eta_g(t)}\right\},$$

where $\tilde{\gamma}_1^2(t) := \min\{N\gamma_1^2(t), \gamma_1(t)\cdot\gamma_2(t)\}$, $\tilde{\gamma}_2^2(t) := \min\{\gamma_2^2(t), N\gamma_1(t)\cdot\gamma_2(t)\}$. All the variables on the RHS can be determined from the continuous-time system, so the maximum sampling interval of the LCFL can be determined.

For cases III–IV, it requires $2C_{41} \le \gamma_1(t), 2C_{42} \le \gamma_2(t)$ and $\{C_{4i}\}_{i=3}^6$ to be positive. It may be difficult to obtain the exact bound for τ_g, τ_ℓ, and Q, but we can derive an approximate bound on these parameters. For $\{C_{4i}\}_{i=3}^6$ to be positive, it requires $\tau_\ell \le \frac{1}{2L\eta_\ell(t)}$, $\tau_g \le \frac{1}{2\eta_g(t)}$. Set $\tau_\ell = \frac{c}{2L\eta_\ell(t)}, \tau_g = \frac{c}{2\eta_g(t)}$ for some $c<1$. By choosing

$$c^2 < \min\left\{\frac{1}{4}, \min\{\tilde{\gamma}_1^2(t), \tilde{\gamma}_2^2(t)\right\} \cdot \min\left\{\frac{1}{L^2\eta_\ell(t)^2\cdot(1+L_f^2)}, \frac{1}{C_g\eta_g^2(t)}\right\}, \tag{3.8}$$

we have $C_{41} = \mathcal{O}(\gamma_1(t)), C_{42} = \mathcal{O}(\gamma_2(t))$. In addition, $Q = \tau_g/\tau_\ell \approx \frac{2L\eta_\ell(t)}{\eta_g(t)}$.

4. Application of the framework. In this section, we discuss some applications of the proposed framework. We first show that by properly choosing the controllers and the discretization scheme, the multirate feedback control system can be specialized to a number of popular distributed algorithms. Due to space limitations, we relegate the discussion of some additional algorithms to Appendix B. Second, we show how the proposed framework can help identify the relationship between different algorithms. Finally, we use DGT as an example to show how the framework can be used to streamline the convergence analysis of a series of algorithms, as well as to facilitate the development of new ones.

4.1. A new interpretation of distributed algorithms. In this part, we map some popular distributed algorithms to the discretized multirate systems, with specific GCFL and LCFL, and specific discretization setting. These mappings together provides a new perspective for understanding distributed algorithms.

Let us begin with mapping the decentralized optimization algorithms.

DGT [35]. The updates are given by

$$\mathbf{x}(k+1) = W\mathbf{x}(k) - c\mathbf{v}(k), \quad \mathbf{v}(k+1) = W\mathbf{v}(k) + \nabla f(\mathbf{x}(k+1)) - \nabla f(\mathbf{x}(k)), \tag{4.1}$$

where $c>0$ is the stepsize. It corresponds to the discretization case III with the following continuous-time controllers:

$$\begin{aligned} &u_{g,x} = (I-W)\cdot\mathbf{x}, \quad u_{g,v} = (I-W)\cdot\mathbf{v}, \\ &u_{\ell,x} = c\mathbf{v}, \quad u_{\ell,v} = -\nabla f(\mathbf{x}) + \nabla f(\mathbf{z}), \quad u_{\ell,z} = \mathbf{z}-\mathbf{x}. \end{aligned} \tag{4.2}$$

NEXT [4]. The updates of NEXT in discrete time are

$$\begin{aligned} \mathbf{x}(k+1/2) &= \arg\min_{\mathbf{x}} \ \tilde{f}(\mathbf{x};\mathbf{x}(k)) + \langle N\mathbf{v}(k) - \nabla f(\mathbf{x}(k)), \mathbf{x}-\mathbf{x}(k)\rangle, \\ \mathbf{x}(k+1) &= W\left(\mathbf{x}(k) + \alpha\cdot(\mathbf{x}(k+1/2) - \mathbf{x}(k))\right), \\ \mathbf{v}(k+1) &= W\mathbf{v}(k) + \nabla f(\mathbf{x}(k+1)) - \mathbf{z}(k), \quad \mathbf{z}(k+1) = \nabla f(\mathbf{x}(k+1)), \end{aligned}$$

where $\tilde{f}$ is some surrogate function; k indicates the iteration index; $\alpha>0$ and $c>0$ are some stepsize parameters. By using the common choice that $\tilde{f}(\mathbf{x};\mathbf{x}(k)) = \langle\nabla f(\mathbf{x}(k)), \mathbf{x}-\mathbf{x}(k)\rangle + \frac{\eta}{2}\|\mathbf{x}-\mathbf{x}(k)\|^2$ (where $\eta>0$ are some constant), the algorithm can be simplify as follows:

$$\begin{aligned} &\mathbf{x}(k+1) = W\mathbf{x}(k) - N\alpha/\eta\cdot\mathbf{v}(k), \quad \mathbf{z}(k+1) = \mathbf{x}(k+1), \\ &\mathbf{v}(k+1) = W\mathbf{v}(k) + \nabla f(\mathbf{x}(k+1)) - \nabla f(\mathbf{z}(k)). \end{aligned} \tag{4.3}$$

Here, $\mathbf{x}$ is the optimization variable, $\mathbf{v}$ tracks the average of the gradients, $\mathbf{z}$ records the one-step-behind state of $\mathbf{x}$. It corresponds to case III, with the continuous-time controllers given by

$$G_g(\mathbf{x},\mathbf{v};A) := \begin{bmatrix} (I-W)\cdot\mathbf{x} \\ (I-W)\cdot\mathbf{v} \end{bmatrix}, \quad G_\ell(x_i,v_i,z_i;f_i) := \begin{bmatrix} v_i \\ \nabla f_i(z_i)-\nabla f_i(x_i) \\ z_i - x_i \end{bmatrix}. \tag{4.4}$$

Next, we discuss two popular federated learning algorithms. In this class of algorithms, the agents are connected with a central server which performs averaging. So the communication graph is a fully connected graph, with the weight matrix being the averaging matrix, i.e., $W = R$, $W_A = I - R$.

FedAvg [1]. The updates are given by (where GD is used for the local steps)

$$\mathbf{x}(k+1) = \begin{cases} R\mathbf{x}(k) - \eta\nabla f(\mathbf{x}(k)), & k \bmod Q = 0, \\ \mathbf{x}(k) - \eta\nabla f(\mathbf{x}(k)), & k \bmod Q \neq 0. \end{cases}$$

This algorithm has the following continuous-time controller:

$$u_{g,x} = \sum_{k=0}^{\infty} \delta(t - k\tau_g)\cdot(I-R)\cdot\mathbf{x}(t), \tag{4.5}$$

where $\delta(t)$ denotes the Dirac delta function. It is interesting to note that FedAvg *cannot* be mapped to a continuous-time double-feedback system, as it does not have a *persistent* GCFL (it is only activated when $t = k\tau_g$; see (4.5)). This partially explains why the FedAvg algorithm requires additional assumptions for convergence.

Scaffold [9]. The updates are given by (where $k_0 := k - (k \bmod K)$)

$$\begin{aligned}
\mathbf{x}(k+1) &= \begin{cases} \mathbf{x}(k) - \eta_1\cdot(\nabla f(\mathbf{x}(k)) - \mathbf{z}(k) + \mathbf{v}(k_0)) - \eta_2\cdot(\mathbf{x}(k) - \mathbf{w}(k)), (k \bmod Q) = 0, \\ \mathbf{x}(k) - \eta_1\cdot(\nabla f(\mathbf{x}(k)) - \mathbf{z}(k) + \mathbf{v}(k_0)),\ (k \bmod Q) \neq 0, \end{cases} \\
\mathbf{v}(k+1) &= \begin{cases} \mathbf{v}(k) - R\cdot(\mathbf{v}(k) + \frac{1}{Q\eta_1}\cdot(\mathbf{w}(k) - \mathbf{x}(k))), & k \bmod Q = 0 \\ \mathbf{v}(k), & k \bmod Q \neq 0, \end{cases} \\
\mathbf{w}(k+1) &= \begin{cases} R\mathbf{x}(k), & k \bmod Q = 0, \\ \mathbf{w}(k), & k \bmod Q \neq 0, \end{cases} \\
\mathbf{z}(k+1) &= \mathbf{z}(k) - \frac{1}{Q}\mathbf{v}(k) - \frac{1}{Q\eta_1}\cdot(\mathbf{x}(k+1) - \mathbf{x}(k)).
\end{aligned}$$

So it uses the discretization case IV. Observe that $\mathbf{w}$ tracks $R\mathbf{x}$, so in continuous-time we have $\mathbf{x} - \mathbf{w} = (I-R)\cdot\mathbf{x} + (R\mathbf{x} - \mathbf{w}) = (I-R)\cdot\mathbf{x} + R\dot{\mathbf{x}}$. Then we can replace $\mathbf{w}$ by $R\cdot(\mathbf{x} - \dot{\mathbf{x}})$, and obtain the continuous-time controller as follows:

$$\begin{aligned}
&u_{g,x} = \eta_2\cdot(I-R)\cdot\mathbf{x} + \eta_1\mathbf{v} + \eta_2 R\dot{\mathbf{x}}, \quad u_{g,v} = -(I-R)\cdot(\mathbf{v} + \dot{\mathbf{x}}/\eta_1), \\
&u_{\ell,x} = \nabla f(\mathbf{x}) - \mathbf{z}, \quad u_{\ell,v} = \mathbf{v} + \dot{\mathbf{x}}/\eta_1, \quad u_{\ell,z} = \mathbf{v} + \dot{\mathbf{x}}/\eta_1.
\end{aligned} \tag{4.6}$$

Finally, we discuss a one rate optimal algorithm.

xFilter [27]. The updates are given by (where $k_0 := k - (k \bmod K)$):

$$\begin{aligned}
\mathbf{x}(k+1) &= \eta_1\cdot((1-\eta_2)I - \eta_2\cdot(I-W))\cdot\mathbf{x}(k) + (1-\eta_1)\cdot\mathbf{x}(k-1) + \eta_1\eta_2\mathbf{v}(k_0) \\
&= \mathbf{x}(k) - \eta_1\eta_2\cdot(2I-W)\mathbf{x}(k) - (1-\eta_1)\cdot(\mathbf{x}(k) - \mathbf{x}(k-1)) + \eta_1\eta_2\mathbf{v}(k_0),
\end{aligned}$$

$$\mathbf{v}(k+1)=\begin{cases}\mathbf{v}(k)+(\mathbf{w}_1(k)-\mathbf{w}_2(k))-(I-W)\cdot\mathbf{x}(k), & k \bmod K=0,\\ \mathbf{v}(k), & k \bmod K\neq 0,\end{cases}$$
$$\mathbf{w}_1(k+1)=\begin{cases}\mathbf{x}(k)-\eta_3\nabla f(\mathbf{x}(k)), & k \bmod K=0,\\ \mathbf{w}_1(k), & k \mod K\neq 0,\end{cases}$$
$$\mathbf{w}_2(k+1)=\begin{cases}\mathbf{w}_1(k), & k \bmod K=0,\\ \mathbf{w}_2(k), & k \bmod K\neq 0,\end{cases}$$

This algorithm uses the discretization case V. We can see $\mathbf{w}_2$ tracks $\mathbf{w}_1$, and $\mathbf{w}_1$ tracks $\mathbf{x}-\eta_3\nabla f(\mathbf{x})$. Therefore, in continuous-time we have $\mathbf{w}_1-\mathbf{w}_2=\dot{\mathbf{x}}-\eta_3\cdot\dot{\nabla} f(\mathbf{x})$, with the following continuous-time system:

$$\begin{aligned}\dot{\mathbf{x}}&=-\eta_1\eta_2\cdot(2I-W)\cdot\mathbf{x}+\eta_1\eta_2\mathbf{v}-(1-\eta_1)\cdot\dot{\mathbf{x}},\\ \dot{\mathbf{v}}&=\dot{\mathbf{x}}-\eta_3\dot{\nabla} f(\mathbf{x})-(I-W)\cdot\mathbf{x}.\end{aligned}\tag{4.7}$$

Integrating over time, and use the initialization that $\mathbf{v}(0)=\mathbf{x}(0)-\eta_3\nabla f(\mathbf{x}(0))$, we have the following expression for $\mathbf{v}(t)$:

$$\mathbf{v}(t)=\int_0^t(\dot{\mathbf{x}}(\tau)-\eta_3\dot{\nabla} f(\mathbf{x}(\tau))-(I-W)\cdot\mathbf{x}(\tau))\mathrm{d}\tau=\mathbf{x}(t)-\eta_3\nabla f(\mathbf{x}(t))-\int_0^t(I-W)\cdot\mathbf{x}(\tau)\mathrm{d}\tau.$$

Define $\mathbf{v}_1=\frac{1}{2-\eta_1}\cdot(\mathbf{x}-\mathbf{v})$, $\mathbf{z}=\frac{\eta_3}{2-\eta_1}\nabla f(\mathbf{x})$, then (4.7) can be equivalently written as

$$\begin{aligned}\dot{\mathbf{x}}&=-\eta_1\eta_2\cdot(I-W)\cdot\mathbf{x}-\eta_1\eta_2\cdot(2-\eta_1)\cdot\mathbf{v}_1-(1-\eta_1)\cdot\dot{\mathbf{x}}\\ &=-\eta_1\eta_2\cdot(I-W)\cdot\mathbf{x}-\eta_1\eta_2\cdot(2-\eta_1)\cdot(\mathbf{v}_1-\mathbf{z})-(1-\eta_1)\cdot\dot{\mathbf{x}}-\eta_1\eta_2\eta_3\nabla f(\mathbf{x}),\\ \dot{\mathbf{v}}_1&=\frac{1}{2-\eta_1}\cdot(I-W)\cdot\mathbf{x}+\frac{\eta_3}{2-\eta_1}\dot{\nabla} f(\mathbf{x}),\quad \dot{\mathbf{z}}=\frac{\eta_3}{2-\eta_1}\dot{\nabla} f(\mathbf{x}).\end{aligned}$$

The dynamic of $\dot{\mathbf{x}}$ implies $\frac{1}{2-\eta_1}(I-R)\cdot(I-W)\cdot\mathbf{x}=-(I-R)\cdot\left(\mathbf{v}_1+\frac{1}{\eta_1\eta_2}\dot{\mathbf{x}}\right)$, where $(I-R)\cdot(I-W)=(I-W)$ by property P1. Substituting this into $\dot{\mathbf{v}}_1$, defining $\eta_4:=\eta_1\eta_2,\eta_5:=(2-\eta_1),\eta_6:=\eta_1\eta_2\eta_3$, and rearranging the terms, we obtain the following equivalent controller:

$$\begin{aligned}&u_{g,x}=\eta_4\cdot(I-W)\cdot\mathbf{x}+\eta_4\eta_5\mathbf{v}_1+(\eta_5-1)\cdot\dot{\mathbf{x}},\quad u_{g,v}=-(I-R)\cdot(\mathbf{v}_1+\dot{\mathbf{x}}/\eta_4),\\ &u_{\ell,x}=\eta_6\nabla f(\mathbf{x})-\eta_4\eta_5\mathbf{z},\quad u_{\ell,v}=\frac{\eta_3}{\eta_5}\dot{\nabla} f(\mathbf{x}),\quad u_{\ell,z}=\frac{\eta_3}{\eta_5}\dot{\nabla} f(\mathbf{x}).\end{aligned}$$

Interestingly, the above dynamics are close to those of Scaffold in (4.6), except that Scaffold uses R instead of W, a different stepsize, and use $R\dot{\mathbf{x}}$ in $u_{g,x}$ instead of $\dot{\mathbf{x}}$.

4.2. Algorithms connections. We summarize the discussion in the previous subsection in Table 2. It is interesting to observe that some seemingly unrelated algorithms, in fact, are very closely related in continuous-time. For example, somewhat surprisingly, Scaffold and xFilter share very similar continuous-time dynamics, although they are designed for very different purposes: the former is designed to improve FedAvg algorithm to better deal with data heterogeneity, while the latter is a primal-dual algorithm designed to achieve the optimal graph dependency. Similarly, each pair of algorithms FedPD and DLM, FedProx and DGD shares the same continuous-time dynamics (these algorithms are discussed in detail in Appendix B). The latter two relations are relatively easier to identify. For example, FedPD and DLM are, in fact, designed from the same primal-dual perspective.

TABLE 2
A summary of the controllers used in different algorithms. In GCFL and LCFL we abstract the most important steps of the controller.

GCFL	LCFL	FL	RO	DO
$(I-W)\cdot\mathbf{x}$	$\nabla f(\mathbf{x})$	FedProx	–	DGD
$(I-W)\cdot\mathbf{y}$	$-\nabla f(\mathbf{x})+\nabla f(\mathbf{z})$	–	–	DGT, NEXT
$c\cdot(I-W)\cdot\mathbf{x}+\mathbf{v}$	$\nabla f(\mathbf{x})$	FedPD	–	DLM
$(I-W)\cdot\mathbf{x}+\eta\mathbf{v}+R\dot{\mathbf{x}}$	$\nabla f(\mathbf{x})-\mathbf{z}$	Scaffold	–	–
$(I-W)\cdot\mathbf{x}+\eta\mathbf{v}+\dot{\mathbf{x}}$	$\nabla f(\mathbf{x})-\mathbf{z}$	–	xFilter	–

Additionally, from the table we can see that there are a few missing entries. Each of these entries represents a new algorithm. Also, we can combine different GCFLs and LCFLs, or design new controllers, to create new control systems (hence algorithms) that are not included in this table.

4.3. Convergence analysis and algorithm design: A case study. In this subsection, we use the DGT algorithm as an example to illustrate how our proposed framework can be used in practice to analyze algorithm behavior, and to facilitate the development of new algorithms.

The iteration of the DGT is given in (4.1). Under A1–A3, this algorithm converges to the stationary point of the problem at a rate of $\mathcal{O}(1/T)$ [18, 28]. To use our framework to analyze it, we will first construct a continuous-time double-feedback system, apply the discretization scheme III, and finally leverage Lemma 3.3 and Theorem 3.4 to obtain the convergence rate.

4.3.1. Continuous-time analysis. We begin by analyzing the continuous-time counterpart of the DGT, whose dynamics, according to (4.2), are given by

$$\begin{aligned}\dot{\mathbf{x}}(t)&=-\eta_g(t)\cdot(I-W)\cdot\mathbf{x}(t)-\eta_\ell(t)\cdot(c\mathbf{v}(t)),\quad \dot{\mathbf{z}}(t)=-\eta_\ell(t)\cdot(\mathbf{z}(t)-\mathbf{x}(t)),\\ \dot{\mathbf{v}}(t)&=-\eta_g(t)\cdot(I-W)\cdot\mathbf{v}(t)+\eta_\ell(t)\cdot(\nabla f(\mathbf{x}(t))-\nabla f(\mathbf{z}(t))),\end{aligned}\tag{4.8}$$

where $\eta_g(t)=1,\quad \eta_\ell(t)=1\,\forall\ t$.

Let us verify properties P1–P5. First, it is easy to prove property P2 with the definition of u_g given in (4.2). To show property P1, recall that we have defined $W:=I-A^T\mathrm{diag}(\mathbf{w})\mathrm{A}$, so it is easy to verify that $\mathbb{1}^T\cdot(I-W)=\mathbb{1}^T\cdot A^T\mathrm{diag}(\mathbf{w})\mathrm{A}=0$ and $C_g=1-\lambda_2(W)$.

To show property P3, we have the following bounds for different parts of the local controller:

$$\begin{aligned}&\|G_{\ell,x}(x_i,v_i,z_i;f_i)-G_{\ell,x}(x'_i,v'_i,z'_i;f_i)\|\\ &\quad=\|c(v_i-v'_i)\|=c\,\|v_i-v'_i\|\,,\\ &\|G_{\ell,v}(x_i,v_i,z_i;f_i)-G_{\ell,v}(x'_i,v'_i,z'_i;f_i)\|\\ &\quad=\|\nabla f_i(x_i)-\nabla f_i(z_i)-\nabla f_i(x'_i)+\nabla f_i(z'_i)\|\\ &\quad\le\|\nabla f_i(x_i)-\nabla f_i(x'_i)\|+\|\nabla f_i(z_i)-\nabla f_i(z'_i)\|\,,\\ &\quad\le L_f(\|x_i-x'_i\|+\|z_i-z'_i\|),\\ &\|G_{\ell,z}(x_i,v_i,z_i;f_i)-G_{\ell,z}(x'_i,v'_i,z'_i;f_i)\|\\ &\quad=\|x_i-z_i-x'_i+z'_i\|\le\|x_i-x'_i\|+\|z_i-z'_i\|\,,\end{aligned}$$

where L_f is the constant of the Lipschitz gradient in A2. So the smoothness constant of the local controller g_ℓ can be expressed as $L=\max\{L_f,c,1\}$.

To verify property P4, let us initialize $\mathbf{v}(t) = \nabla f(\mathbf{x}(t)), \mathbf{z}(t) = \mathbf{x}(t)$, and assume that $\eta_g(t) = 0$ in (4.8), that is, the GCFL is inactive. Then we have

$$\begin{aligned} &\mathbf{z}(t+\tau) = \mathbf{x}(t+\tau),\ \mathbf{v}(t+\tau) = \nabla f(\mathbf{x}(t+\tau)), \\ &\dot{\mathbf{x}}(t+\tau) = -c\mathbf{v}(t+\tau) = -c\nabla f(\mathbf{x}(t+\tau)). \end{aligned} \tag{4.9}$$

Further, we can verify that the output of the LCFL can be bounded by

$$\begin{aligned} &\|u_{i,\ell,x}(t)\| = \|c \cdot v_i(t)\| = c\,\|\nabla f_i(x_i(t))\|\,, \\ &\|u_{i,\ell,v}(t)\| = \|\nabla f_i(x_i(t)) - \nabla f_i(z_i(t))\| \le 2\,\|\nabla f_i(x_i(t))\|\,, \\ &\|u_{i,\ell,z}(t)\| = \|z_i(t) - x_i(t)\| = \|c \cdot v_i(t)\| = c\,\|\nabla f_i(x_i(t))\|\,. \end{aligned}$$

The algorithm becomes the gradient flow algorithm that satisfies property P4 with $\alpha(t) = c$, $C_x = c, C_v \le 2, C_z = c$. Finally, we verify property P5. We can compute $\dot{\mathcal{E}}(t)$ as follows:

$$\begin{aligned} \dot{\mathcal{E}}(t) &= -\left\langle \nabla f(\bar{\mathbf{x}}(t)), \frac{1}{N}\sum_{i=1}^{N} u_{\ell,x}(t) \right\rangle - \langle (I-R)\cdot \mathbf{y}(t), u_{g,y}(t) + u_{\ell,y}(t)\rangle \\ &\overset{(4.2)}{=} -\langle \nabla f(\bar{\mathbf{x}}(t)), c\bar{\mathbf{v}}(t)\rangle - \langle (I-R)\cdot \mathbf{y}(t), (I-W)\cdot \mathbf{y}(t)\rangle \\ &\quad - \langle (I-R)\cdot \mathbf{x}(t), c\mathbf{v}(t)\rangle + \langle (I-R)\cdot \mathbf{v}(t), \nabla f(\mathbf{x}(t)) - \nabla f(\mathbf{z}(t))\rangle\,. \end{aligned} \tag{4.10}$$

Then we bound each term on the RHS above separately, and finally integrate. The detailed derivation is relegated to [37, sec. D]. The final bound we can obtain is

$$\begin{aligned} \int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le &-\frac{c}{2}\int_0^t \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2\,\mathrm{d}\tau - \frac{c - 8L_f c^2/\beta}{2}\int_0^t \|\bar{\mathbf{v}}(\tau)\|^2\,\mathrm{d}\tau \\ &- \Big(C_g - \frac{c + 2cL_f + \beta + 16cL_f/\beta}{2}\Big)\cdot \int_0^t \|(I-R)\cdot \mathbf{y}(\tau)\|^2\,\mathrm{d}\tau. \end{aligned}$$

By choosing $\beta < C_g/2$, $\frac{C_g^2}{64L_f} \le c \le \frac{C_g^2}{32L_f}$, we can verify that the dynamics of the continuous-time system (4.8) satisfy (2.8), with $\gamma_1(t) \ge \frac{C_g^2}{128L_f}$ and $\gamma_2(t) \ge \frac{C_g}{4}$. Applying Theorem 2.1, we know that continuous-time gradient tracking algorithm converges in $\mathcal{O}(1/T)$.

4.3.2. New algorithm design. Now that we have verified properties P1–P5 for the continuous-time system (4.8), we can derive a number of related algorithms by adjusting the discretization schemes, or by changing the GCFL.

Let us first consider changing the discretization scheme from cases III to IV, where $\tau_g = Q\tau_\ell > 0$. In this case, there will be Q local computation steps between every two communication steps. This kind of update scheme is closely related to algorithms in FL, and we refer to the resulting algorithm the Decentralized Federated Gradient Tracking (D-FedGT) algorithm. Its steps are listed below (where $k_0 = k - (k \bmod Q)$):

$$\begin{aligned} &\mathbf{x}(k+1) = \mathbf{x}(k) - \tau_\ell \mathbf{v}(k) - \tau_g(I-W)\mathbf{x}(k_0), \\ &\mathbf{v}(k+1) = \mathbf{v}(k) + \nabla f(\mathbf{x}(k+1)) - \nabla f(\mathbf{x}_k) - \tau_g(I-W)\mathbf{v}(k_0). \end{aligned} \tag{4.11}$$

By applying Lemma 3.3 and Theorem 3.4, we can directly obtain that this new algorithm also converges with rate $\mathcal{O}(\frac{1}{T})$ with properly chosen constant τ_ℓ, τ_g and Q following Lemma 3.3 and (3.8).

Second, we can replace the GCFL of the DGT with an *accelerated* consensus controller [7]. This leads to a new Accelerated Gradient Tracking (AGT) algorithm:

$$\begin{aligned}
&\mathbf{x}(k+1) = \mathbf{x}(k) - \eta'_\ell \mathbf{v}(k) - \eta'_g(1+c)\mathbf{x}(k) + c\mathbf{v}_x(k),\\
&\mathbf{v}(k+1) = \mathbf{v}(k) + \nabla f(\mathbf{x}(k+1)) - \nabla f(\mathbf{x}(k)) - \eta_g(1+c)\mathbf{v}(k) + c\mathbf{v}_v(k),\\
&\mathbf{v}_x(k+1) = \mathbf{x}(k), \quad \mathbf{v}_v(k+1) = \mathbf{v}(k), \ \text{ where } c := \frac{1-\sqrt{1-\lambda_2(W)}}{1+\sqrt{1-\lambda_2(W)^2}}.
\end{aligned} \tag{4.12}$$

Then by examining property P1, we know that the network dependency of the new algorithm improved from C_g to $\hat{C}_g = C_g \cdot \frac{\sqrt{C_g}+\sqrt{2-C_g}}{\sqrt{C_g+C_g\sqrt{2-C_g}}} > C_g$. And when C_g is small, $\hat{C}_g$ scales with $\sqrt{C_g}$. Then according to the derivation in the last subsection, we have $\gamma_2(t) \geq \frac{\hat{C}_g}{4}$. Finally, we can apply Theorem 3.4, and assert that the new algorithm improves the convergence speed from $\mathcal{O}(\frac{1}{C_g T})$ to $\mathcal{O}(\frac{1}{\hat{C}_g T})$.

4.3.3. Numerical results. We provide numerical results for implementations of Continuous-time (CT) DGT, the D-FedGT, and D-AGT algorithms discussed in the previous subsection. We first verify an observation from Theorem 3.4, that discretization slows down the convergence speed of the system. Towards this end, we conduct numerical experiments with different discretization patterns and compare the convergence speed in terms of the stationarity gap. Then we compare the convergence speed of CT-DGT and CT-AGT to demonstrate the benefit of changing the controller in the GCFL from the standard consensus controller to the accelerated one.

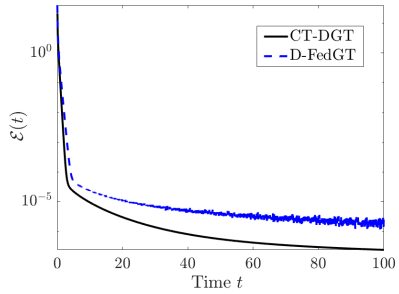
In the experiments, we consider the nonconvex regularized logistic regression problem:

$$f_i(\mathbf{x}; (\mathbf{a}_i, b_i)) = \log(1+\exp(-b_i \mathbf{x}^T \mathbf{a}_i)) + \sum_{d=1}^{d_x} \frac{\beta\alpha(\mathbf{x}[d])^2}{1+\alpha(\mathbf{x}[d])^2},$$

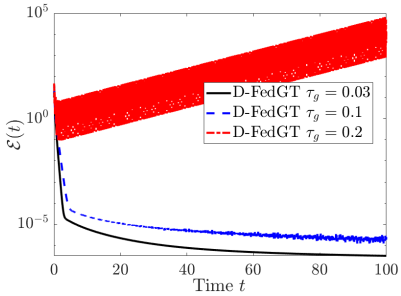
where $\mathbf{a}_i$ denotes the features and b_i denotes the labels of the dataset on the ith agent. We set the number of agent $N = 20$ and each agent has a local dataset of size 500. We use an Erdős–Rényi random graph with density 0.5 for the network and optimize the weight matrix W to achieve the optimal C_g. We set $c = 1$ for the gradient tracking algorithm.

We first compare CT-DGT ($\tau_g = \tau_\ell = 0$) and D-FedGT ($\tau_g = 0.1, \tau_\ell = 0.005, Q = 20$), the result of CT-DGT and D-FedGT is shown in Figure 4a. We can see that by discretizing each loop, the system converges slower as compared with the CT system. Figure 4b shows the convergence behavior of the D-FedGT algorithm with different τ_g. We observe that by increasing the sampling interval for GCFL, the convergence of the system slows down and it eventually diverges. Figures 4c and 4d show the convergence results of D-AGT compared with DGT in both CT and in case III. We observe that by changing the GCFL, D-AGT converges faster than DGT.

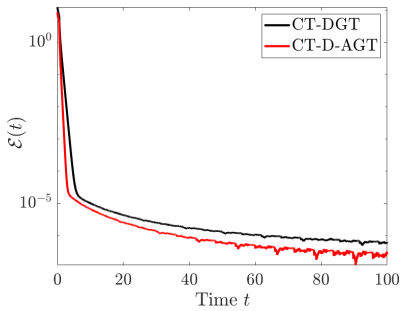
5. Conclusion. In this work, we have designed a framework to understand distributed optimization algorithms from a control perspective. We have shown that a multirate double-feedback control system can represent a wide range of deterministic distributed optimization algorithms. We use a few examples to demonstrate how the proposed framework can help understand the connection between algorithms, as well as facilitate new algorithm design. In the future, we plan to extend the framework to model distributed stochastic algorithms.



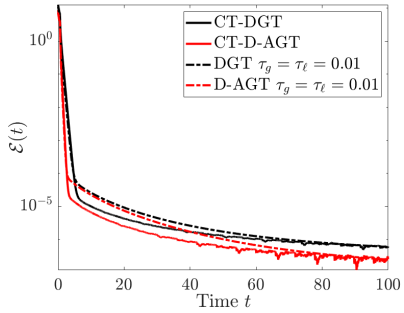
(a) The evoluation of the Energy function $\mathcal{E}(t)$ of CT-CGT, D-FedGT.



(b) Energy function $\mathcal{E}(t)$ of D-FedGT with different intervals τ_g.



(c) The evolution of the Energy function $\mathcal{E}(t)$ of CT-DGT and CT-D-AGT.



(d) The evolution of the Energy function $\mathcal{E}(t)$ of DGT and D-AGT.

FIG. 4. *The performance of Continuous-GT, D-FedGT, D-MGT, and AGT.*

Appendix A. Proofs of Section 3. Let t_ℓ (resp., t_g) denote the time at which the local (resp., global) controller samples, that is, $t_\ell := t - t \bmod \tau_\ell$ and $t_g := t - t \bmod \tau_g$. To simplify the analysis, we treat the stepsizes $\eta_\ell(t), \eta_g(t)$ as constants in each sampling intervals. Also recall that $\mathbf{y}(t) = [\mathbf{x}(t); \mathbf{v}(t)]$. The following relations will be useful:

$$\langle a, b\rangle = \frac{1}{2\alpha}\|a\|^2 + \frac{\alpha}{2}\|b\|^2 - \frac{1}{2}\left\|\frac{1}{\sqrt{\alpha}}a + \sqrt{\alpha}b\right\|^2 \le \frac{1}{2\alpha}\|a\|^2 + \frac{\alpha}{2}\|b\|^2, \tag{A.1}$$

$$(I-R)^2 = I - 2R + R^2 = I - R, \quad \|R\| \le 1, \quad \|I-R\| \le 1. \tag{A.2}$$

The proofs of Lemmas 3.1, 3.2, and 3.3 adopt the similar concept in robust control theory. The time derivative of the energy function of the discretized system is given by

$$\dot{\mathcal{E}}(t) = \underbrace{-\left\langle \nabla f(\bar{\mathbf{x}}(t)), \frac{1}{N}\mathbb{1}^T \eta_\ell(t) u_{\ell,x}(t)\right\rangle - \langle (I-R)\cdot \mathbf{y}(t), \eta_\ell(t)\cdot u_{\ell,y}(t) + \eta_g(t)\cdot u_g(t)\rangle}_{\text{term I}} + \hat{\mathcal{E}}(t), \tag{A.3}$$

where "term I" is the derivative of the CT energy function given in (2.7); $\hat{\mathcal{E}}(t)$ is the error caused by discretization. By integrating (A.3) and apply property P5, we have

$$\int_0^t \dot{\hat{\mathcal{E}}}(t) \le -\int_0^t \gamma_1(\tau)\left\|\nabla f(\bar{\mathbf{x}}(\tau))\right\|^2 + \gamma_2(\tau)\left\|(I-R)\cdot\mathbf{y}(\tau)\right\|^2 \mathrm{d}\tau + \int_0^t \hat{\mathcal{E}}(\tau)\mathrm{d}\tau. \tag{A.4}$$

The key idea of proofs is to bound $\int_0^t \hat{\mathcal{E}}(\tau)d\tau$ by the first two terms.

A.1. Proof of Lemma 3.1. In this case $\hat{u}_g(t) = G_g(\mathbf{x}(t_g), \mathbf{v}(t_g); A)$. By taking derivative of $\mathcal{E}(t)$, and by comparing with (A.3), we can obtain

$$\hat{\mathcal{E}}(t) = \eta_g(t)\left\langle (I-R)\cdot\mathbf{y}(t), u_g(t) - \hat{u}_g(t)\right\rangle. \tag{A.5}$$

Next, we bound $\int_0^t \hat{\mathcal{E}}(\tau)d\tau$. Towards this end, we first observe that

$$\begin{aligned}
\left\langle (I-R)\cdot\mathbf{y}(t), u_g(t) - \hat{u}_g(t)\right\rangle &\overset{(i)}{=} \left\langle (I-R)\cdot\mathbf{y}(t), G_g(\mathbf{y}(t) - \mathbf{y}(t_g); A)\right\rangle \\
&= \left\langle (I-R)\cdot\mathbf{y}(t), G_g\Big(\int_{t_g}^t \dot{\mathbf{y}}(s)ds; A\Big)\right\rangle \\
&\overset{(A.1)}{\le} \frac{\gamma_2(t)}{2}\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2 \\
&\quad + \frac{1}{2\gamma_2(t)}\left\|G_g\Big(\int_{t_g}^t \dot{\mathbf{y}}(s)ds; A\Big)\right\|^2,
\end{aligned}$$

where (i) is due to the linearity property P2. Next, we bound the last term above by $\|\nabla f(\bar{\mathbf{x}}(t))\|^2$ and $\|(I-R)\cdot\mathbf{y}(t)\|^2$. To proceed, let us define

$$\begin{aligned}
&\tilde{\mathbf{y}}(t) := G_g\Big(\int_{t_g}^t \dot{\mathbf{y}}(s)ds; A\Big) = u_g(t) - \hat{u}_g(t), \quad \mathbf{w}(t) := [(I-R)\cdot\mathbf{y}(t); \nabla f(\bar{\mathbf{x}}(t))], \\
&q(t) := \left\|G_g\Big(\int_{t_g}^t \dot{\mathbf{y}}(s)ds; A\Big)\right\| / \left\|[(I-R)\cdot\mathbf{y}(t); \nabla f(\bar{\mathbf{x}}(t))]\right\| = \|\tilde{\mathbf{y}}(t)\|/\|\mathbf{w}(t)\|.
\end{aligned} \tag{A.6}$$

Using the above definition, we have

$$\left\|G_g\Big(\int_{t_g}^t \dot{\mathbf{y}}(s)ds; A\Big)\right\|^2 = \|\tilde{\mathbf{y}}(t)\|^2 = q^2(t)\left\|\mathbf{w}(t)\right\|^2. \tag{A.7}$$

It then suffices to bound $q(t)$. Towards this end, let us first bound $\|\dot{\mathbf{w}}(t)\|$ by

$$\begin{aligned}
\|\dot{\mathbf{w}}(t)\| &\overset{(i)}{=} \left\|\left[(I-R)\cdot(\eta_g(t)\hat{u}_g(t) + \eta_\ell(t)u_{\ell,y}(t)); \left\langle \partial^2 f(\bar{\mathbf{x}}(t)), \eta_\ell(t)\frac{\mathbb{1}^T}{N}u_{\ell,x}(t)\right\rangle\right]\right\| \\
&\le \eta_g(t)\left\|(I-R)\cdot\hat{u}_g(t)\right\| + \min\left\{\eta_\ell(t), \frac{\eta_\ell(t)\left\|\partial^2 f(\bar{\mathbf{x}}(t))\right\|}{N}\right\}\|u_{\ell,y}(t)\| \\
&\overset{(ii)}{\le} \eta_g(t)\left(\left\|(I-R)\cdot(u_g(t) - \hat{u}_g(t))\right\| + \left\|(I-R)\cdot u_g(t)\right\|\right) \\
&\quad + \sqrt{C_x^2 + C_v^2}\cdot\eta_\ell(t)\cdot\Big(1 + \frac{L_f}{N}\Big)\cdot\|\nabla f(\mathbf{x}(t))\| \\
&\overset{(iii)}{\le} \eta_g(t)\left(\|\tilde{\mathbf{y}}(t)\| + \|(I-R)\cdot\mathbf{y}(t)\|\right) + \sqrt{C_x^2 + C_v^2}\cdot\eta_\ell(t)\cdot\Big(1 + \frac{L_f}{N}\Big)\cdot\|\nabla f(\mathbf{x}(t))\|
\end{aligned}$$

$$
\begin{aligned}
&\overset{\text{(iv)}}{\leq} \eta_g(t)\cdot q(t)\cdot\|\mathbf{w}(t)\| + \eta_g(t)\cdot\|(I-R)\cdot\mathbf{y}(t)\| \\
&\qquad + \sqrt{C_x^2+C_v^2}\cdot\eta_\ell(t)\cdot\left(1+\frac{L_f}{N}\right)\cdot\left(\|\nabla f(\bar{\mathbf{x}}(t))\| + \frac{L_f}{N}\|(I-R)\cdot\mathbf{x}(t)\|\right) \\
&\overset{\text{(v)}}{\leq} \sqrt{2}\left(\eta_g(t)q(t)+\eta_g(t)+\sqrt{C_x^2+C_v^2}\cdot\eta_\ell(t)\cdot\left(1+\frac{L_f}{N}\right)^2\right)\cdot\|\mathbf{w}(t)\|,
\end{aligned}
\tag{A.8}
$$

where (i) can be derived similarly as in (2.7); in (ii) we add and subtract $u_g(t)$ to the first term, apply property P4 to the last term, used the following definition of sub-Hessian:

$$
\lim_{\delta\to 0}\frac{\left\|f(x+\delta)-f(x)-\langle\nabla f(x),\delta\rangle-\frac{1}{2}\delta^T\partial^2 f(x)\delta\right\|}{\|\delta\|^2}=0,
$$

and the fact that that under the smoothness A2, it holds that $\|\partial^2 f(\mathbf{x})\|\leq L$ [23, Theorem 3.1]; in (iii) we combine $\|I-R\|\leq 1$ and (2.1) to the second term, use the definition of $\tilde{\mathbf{y}}(t)$ in (A.6); in (iv) we use the definition of $q(t)$ in (A.6), add and subtract $\nabla f(\bar{\mathbf{x}}(t))$ to the last term and apply A2; in (v) we use the fact that $\|a\|+\|b\|\leq\sqrt{2(\|a\|^2+\|b\|^2)}$, and $\mathbf{x}$ is a subvector of $\mathbf{y}$. Then we can bound $\dot{q}(t)$ by

$$
\begin{aligned}
\dot{q}(t) &= \frac{\dot{\tilde{\mathbf{y}}}(t)^T\tilde{\mathbf{y}}(t)}{\|\mathbf{w}(t)\|\,\|\tilde{\mathbf{y}}(t)\|} - \frac{\|\tilde{\mathbf{y}}(t)\|\,\mathbf{w}(t)^T\dot{\mathbf{w}}(t)}{\|\mathbf{w}(t)\|^3} \\
&\overset{\text{(i)}}{\leq} \frac{\left\|\dot{\tilde{\mathbf{y}}}(t)\right\|\|\tilde{\mathbf{y}}(t)\|}{\|\mathbf{w}(t)\|\,\|\tilde{\mathbf{y}}(t)\|} + \frac{\|\tilde{\mathbf{y}}(t)\|\,\|\mathbf{w}(t)\|\,\|\dot{\mathbf{w}}(t)\|}{\|\mathbf{w}(t)\|^3} \overset{\text{(ii)}}{\leq} (1+q(t))\frac{\|\dot{\mathbf{w}}(t)\|}{\|\mathbf{w}(t)\|} \\
&\overset{\text{(A.8)}}{\leq} (1+q(t))\cdot\sqrt{2}\left(q(t)\eta_g(t)+\eta_g(t)+\sqrt{C_x^2+C_v^2}\eta_\ell(t)\cdot\left(1+\frac{L_f}{N}\right)^2\right),
\end{aligned}
$$

where in (i) we apply the Cauchy–Schwarz inequality; (ii) is due to the definition of $q(t)$ in (A.6), and the relations below (where equality comes from the linearity property P2):

$$
\left\|\dot{\tilde{\mathbf{y}}}(t)\right\| = \|G_g(\dot{\mathbf{y}}(t);A)\| \overset{(2.1)}{\leq} \|(I-R)\cdot\dot{\mathbf{y}}(t)\| \leq \|\dot{\mathbf{w}}(t)\|.
$$

Note that $q(t_g)=0$, solve the above inequality of $\dot{q}(t)$ by using Grownwall's inequality, we obtain $q(t)\leq q_{\max}:=\exp\{\sqrt{2}\tau_g\cdot(\sqrt{C_x^2+C_v^2}\cdot\eta_\ell(t)\cdot(1+L_f/N)^2)\}-1$. Plug in this estimate to (A.7), and further to (A.5) and (A.4), we obtain

$$
\begin{aligned}
\int_0^t\dot{\mathcal{E}}(\tau)\mathrm{d}\tau &\leq \int_0^t\left(-\gamma_1(\tau)\|\nabla f(\bar{\mathbf{x}}(\tau))\|^2-\gamma_2(\tau)\|(I-R)\cdot\mathbf{y}(\tau)\|^2\right)\mathrm{d}\tau \\
&\quad+\int_0^t\left(\frac{\gamma_2(\tau)}{2}\|(I-R)\cdot\mathbf{y}(\tau)\|^2+\frac{1}{2\gamma_2(\tau)}q_{\max}^2\|\mathbf{w}(\tau)\|^2\right)\mathrm{d}\tau \\
&=\int_0^t-\left(\gamma_1(\tau)-\frac{q_{\max}^2}{2\gamma_2(\tau)}\right)\cdot\|\nabla f(\bar{\mathbf{x}}(\tau))\|^2-\left(\frac{\gamma_2(\tau)}{2}-\frac{q_{\max}^2}{2\gamma_2(\tau)}\right)\cdot\|(I-R)\cdot\mathbf{y}(\tau)\|^2\mathrm{d}\tau.
\end{aligned}
$$

A.2. Proof of Lemma 3.2. For notation simplicity, let us define the discrete time controller output as $\hat{u}_{i,\ell}(t)=G_{i,\ell}(x_i(t_\ell),v_i(t_\ell),z_i(t_\ell);f_i)$. Then we can write $\dot{\mathcal{E}}(t)$ similarly as in (A.3), and the error term $\hat{\mathcal{E}}(t)$ in this case can be expressed, and

bounded as below:

$$\begin{aligned}\hat{\mathcal{E}}(t) &= \left\langle \nabla f(\bar{\mathbf{x}}(t)), \frac{\eta_\ell(t)}{N}\mathbb{1}^T(u_{\ell,x}(t)-\hat{u}_{\ell,x}(t))\right\rangle \\ &\quad + \langle (I-R)\mathbf{y}(t), \eta_\ell(t)(I-R)\cdot(u_{\ell,y}(t)-\hat{u}_{\ell,y}(t))\rangle \\ &\overset{(A.1)}{\leq} \frac{\gamma_1(t)}{2}\left\|\nabla f(\bar{\mathbf{x}}(t))\right\|^2 + \frac{\gamma_2(t)}{2}\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2 \\ &\quad + \frac{\eta_\ell^2(t)}{2N\gamma_1(t)}\left\|R\cdot(u_{\ell,y}(t)-\hat{u}_{\ell,y}(t))\right\|^2 + \frac{\eta_\ell^2(t)}{2\gamma_2(t)}\left\|(I-R)\cdot(u_{\ell,y}(t)-\hat{u}_{\ell,y}(t))\right\|^2 \\ &\leq \frac{\gamma_1(t)}{2}\left\|\nabla f(\bar{\mathbf{x}}(t))\right\|^2 + \frac{\gamma_2(t)}{2}\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2 \\ &\quad + \frac{\eta_\ell^2(t)L^2}{2\min\{N\gamma_1(t),\gamma_2(t)\}}\left(\left\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\right\|^2 + \left\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\right\|^2\right), \end{aligned} \tag{A.9}$$

where the last inequality combines (A.2) and the Lipschitz gradient property P3, which gives

$$\begin{aligned}\left\|u_{\ell,y}(t)-\hat{u}_{\ell,y}(t)\right\|^2 &= \sum_{i=1}^N \left\|G_\ell(\mathbf{x}_i(t),\mathbf{v}_i(t),\mathbf{z}_i(t)) - G_\ell(\mathbf{x}_i(t_\ell),\mathbf{v}_i(t_\ell),\mathbf{z}_i(t_\ell))\right\|^2 \\ &\leq L^2(\left\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\right\|^2 + \left\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\right\|^2).\end{aligned}$$

The key step is to bound the last term in (A.9). Towards this end, first note that we have the following relations from (3.3) and property P2:

$$\begin{aligned}(I-R)\cdot\dot{\mathbf{y}}(t) &= -\eta_g(t)\cdot(I-R)\cdot u_{g,y}(t) - \eta_\ell(t)\cdot(I-R)\cdot\hat{u}_{\ell,y}(t) \\ &= -\eta_g(t)\cdot(I-R)\cdot W_A\mathbf{y}(t) - \eta_\ell(t)\cdot(I-R)\cdot\hat{u}_{\ell,y}(t).\end{aligned}$$

Solving this differential equation with initial condition $\mathbf{y}(t_\ell)$, we obtain

$$\begin{aligned}&(I-R)\cdot\mathbf{y}(t) \\ &= \mathrm{e}^{-(I-R)\cdot W_A\int_{t_\ell}^t \eta_g(s)\mathrm{d}s}\left(\mathbf{y}(t_\ell) - \int_{t_\ell}^t \eta_\ell(s)\mathrm{e}^{(I-R)\cdot W_A\int_{t_\ell}^s \eta_g(s_1)\mathrm{d}s_1}\mathrm{d}s\cdot\hat{u}_{\ell,y}(t)\right).\end{aligned} \tag{A.10}$$

This expression for $\mathbf{y}(t_\ell)$ can be used to further bound the following term:

$$\begin{aligned}&\left\|(I-R)\cdot(\mathbf{y}(t)-\mathbf{y}(t_\ell))\right\|^2 \\ &\overset{(A.10)}{=} \left\|(I-R)\cdot\left(\mathbf{y}(t) - \left(\mathrm{e}^{-(I-R)\cdot W_A\int_{t_\ell}^t \eta_g(s)\mathrm{d}s}\right)^{-1}(I-R)\cdot\mathbf{y}(t)\right.\right. \\ &\qquad \left.\left. - \int_{t_\ell}^t \eta_\ell(s)\mathrm{e}^{(I-R)\cdot W_A\int_{t_\ell}^s \eta_g(s_1)\mathrm{d}s_1}\mathrm{d}s\cdot\hat{u}_{\ell,y}(t)\right)\right\|^2 \\ &\overset{(i)}{\leq} (1+\beta)\left\|I-(I-R)\cdot\left(\mathrm{e}^{-(I-R)\cdot W_A\int_{t_\ell}^t \eta_g(s)\mathrm{d}s}\right)^{-1}\right\|^2\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2 \\ &\qquad + \left(1+\frac{1}{\beta}\right)\left\|\int_{t_\ell}^t \eta_\ell(s)\mathrm{e}^{(I-R)\cdot W_A\int_{t_\ell}^s \eta_g(s_1)\mathrm{d}s_1}\mathrm{d}s\cdot(I-R)\cdot\hat{u}_{\ell,y}(t))\right\|^2\end{aligned} \tag{A.11}$$

$$\begin{aligned}
&\overset{\text{(ii)}}{\leq} (1+\beta)\cdot\left(\frac{1-C_y}{C_y}\right)^2\cdot\|(I-R)\cdot\mathbf{y}(t)\|^2\\
&\qquad+\left(1+\frac{1}{\beta}\right)\cdot\left(\frac{\tau_\ell\eta_\ell(t)}{C_y}\right)^2\cdot\|(I-R)\cdot\hat{u}_{\ell,y}(t))\|^2\\
&\overset{\text{(iii)}}{=}\left(\frac{1-C_y}{C_y^2}\right)\cdot\|(I-R)\cdot\mathbf{y}(t)\|^2+\left(\frac{\tau_\ell^2\eta_\ell^2(t)}{C_y}\right)\cdot\|(I-R)\cdot\hat{u}_{\ell,y}(t))\|^2,
\end{aligned}\tag{A.12}$$

where in (i) we use Cauchy–Schwarz inequality (with $\beta>0$ being an arbitrary constant); in (ii) we bound the first norm with property P1 so that $\|(I-R)W_A\| = \|W_A\|\geq C_g$, which implies the following:

$$\left\|I-(I-R)\cdot\left(\mathrm{e}^{-(I-R)\cdot W_A\int_{t_\ell}^t\eta_g(s)\mathrm{d}s}\right)^{-1}\right\|^2\leq\left(1-(\mathrm{e}^{-C_g\int_{t_\ell}^t\eta_g(s)\mathrm{d}s})^{-1}\right)^2;$$

then by using the fact that $t-t_\ell\leq\tau_\ell$, $\eta_g(s)$ can be treated as constant in the integration, and define $C_y:=\mathrm{e}^{-C_g\tau_\ell\eta_g(t)}$, the bound can be further simplified as $\left(1-(\mathrm{e}^{-C_g\int_{t_\ell}^t\eta_g(s)\mathrm{d}s})^{-1}\right)^2\leq\left(1-\frac{1}{C_y}\right)^2$; in (iii) we choose $\beta=\frac{C_y}{1-C_y}$.

Using the system dynamics (3.3), we have

$$R\cdot\mathbf{y}(t)=R\cdot\mathbf{y}(t_\ell)-\left(\int_{t_\ell}^t\eta_\ell(s)\mathrm{d}s\right)R\hat{u}_{\ell,y}(t).\tag{A.13}$$

Then we can bound the last term of (A.9) by

$$\begin{aligned}
&\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\|^2+\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\|^2\\
&\overset{\text{(i)}}{=}\|(I-R)\cdot(\mathbf{y}(t)-\mathbf{y}(t_\ell))\|^2+\|R\cdot(\mathbf{y}(t)-\mathbf{y}(t_\ell))\|^2+\left\|\int_{t_\ell}^t\eta_\ell(s)\mathrm{d}s\right\|^2\cdot\|\hat{u}_{\ell,z}(t)\|^2\\
&\overset{\text{(A.12),(A.13)}}{\leq}\left(\frac{1-C_y}{C_y^2}\right)\cdot\|(I-R)\cdot\mathbf{y}(t)\|^2+\left(\frac{\tau_\ell^2\eta_\ell^2(t)}{C_y}\right)\cdot\|(I-R)\cdot\hat{u}_{\ell,y}(t)\|^2\\
&\qquad+\left\|\int_{t_\ell}^t\eta_\ell(s)\mathrm{d}s\right\|^2\|R\hat{u}_{\ell,y}(t)\|^2+\left\|\int_{t_\ell}^t\eta_\ell(s)\mathrm{d}s\right\|^2\cdot\|\hat{u}_{\ell,z}(t)\|^2\\
&\overset{\text{(ii)}}{\leq}\left(\frac{1-C_y}{C_y^2}\right)\cdot\|(I-R)\cdot\mathbf{y}(t)\|^2+\frac{(\tau_\ell\eta_\ell(t))^2}{\min\{C_y,1\}}\left(\|\hat{u}_{\ell,y}(t)\|^2+\|\hat{u}_{\ell,z}(t)\|^2\right)\\
&\overset{\text{(iii)}}{\leq}\left(\frac{1-C_y}{C_y^2}\right)\cdot\|(I-R)\cdot\mathbf{y}(t)\|^2+2C_\ell^2\left(\|u_\ell(t)-\hat{u}_\ell(t)\|^2+\|u_\ell(t)\|^2\right)\\
&\overset{\text{(iv)}}{\leq}\left(\frac{1-C_y}{C_y^2}\right)\cdot\|(I-R)\cdot\mathbf{y}(t)\|^2+2L^2C_\ell^2\left(\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\|^2+\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\|^2\right)\\
&\qquad+4C_\ell^2\cdot(C_x^2+C_v^2+C_z^2)\cdot(\|\nabla f(\bar{\mathbf{x}}(t))\|^2+\|\nabla f(\mathbf{x}(t))-\nabla f(\bar{\mathbf{x}}(t))\|^2)\\
&\overset{\text{(v)}}{\leq}\frac{\left(\frac{1-C_y}{C_y^2}\right)+4L_f^2C_\ell^2C_f}{1-2L^2C_\ell^2}\|(I-R)\cdot\mathbf{y}(t)\|^2+\frac{4C_\ell^2C_f}{1-2L^2C_\ell^2}\|\nabla f(\bar{\mathbf{x}}(t))\|^2,
\end{aligned}\tag{A.14}$$

where in (i) we separate $\mathbf{y}(t)-\mathbf{y}(t_\ell)$ into $R\cdot(\mathbf{y}(t)-\mathbf{y}(t_\ell))+(I-R)\cdot(\mathbf{y}(t)-\mathbf{y}(t_\ell))$, expand the square, and use the fact that $R\cdot(I-R)=0$; in (ii) we bound the integration

interval in the last two terms with $t - t_\ell \le \tau_\ell$, using the fact that $\eta_\ell(s)$ is treated as constant in the integration, and combine the last three terms; in (iii) we add and subtract $u_\ell(t)$ to the last term and apply the Cauchy–Schwarz inequality and further define $C_\ell := \frac{\tau_\ell \eta_\ell(t)}{\min\{C_y,1\}}$; in (iv) we apply properties P3 and P4 to the last two terms and define

$$C_f := C_x^2 + C_v^2 + C_z^2; \tag{A.15}$$

in (v) we apply A2 to the last term and move $\|\mathbf{y}(t) - \mathbf{y}(t_\ell)\|^2 + \|\mathbf{z}(t) - \mathbf{z}(t_\ell)\|^2$ to the left and divide both sides by $1 - 2L^2C_\ell^2$ (note that this operation is legitimate since we have chosen $\tau_\ell \le \frac{1+2C_g\eta_g(t)}{2L\eta_\ell(t)}$ such that $2L^2C_\ell^2 < 1$).

Substitute to $\hat{\mathcal{E}}$ in (A.4), we have

$$\int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le \int_0^t \left(-\left(\frac{\gamma_1(\tau)}{2} - C_{21}\right)\|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 - \left(\frac{\gamma_2(\tau)}{2} - C_{22}\right)\|(I-R)\cdot\mathbf{y}(\tau)\|^2 \right)\mathrm{d}\tau,$$

where $C_{21} := \frac{4L^2C_\ell^2\eta_\ell^2(\tau)\cdot C_f}{2(1-2L^2C_\ell^2)\cdot\min\{N\gamma_1(\tau),\gamma_2(\tau)\}}$ and $C_{22} := \frac{L^2\eta_\ell^2(\tau)\cdot\left(\left(\frac{1-C_y}{C_y^2}\right)+4L_f^2C_\ell^2C_f\right)}{2(1-2L^2C_\ell^2)\cdot\min\{N\gamma_1(\tau),\gamma_2(\tau)\}}$.

A.3. Proof for Lemma 3.3. In cases III–IV, we have $\tau_g = Q\tau_\ell$. Also note that t_g, t_ℓ were defined at the beginning of Appendix A. The update of the states can be written as

$$\begin{aligned}
\mathbf{y}(t_g + (q+1)\tau_\ell) &= \mathbf{y}(t_g + q\tau_\ell) - \int_{t_g+q\tau_\ell}^{t_g+(q+1)\tau_\ell} \eta_g(s)\hat{u}_g(s) + \eta_\ell(s)\hat{u}_{\ell,y}(s)\mathrm{d}s,\\
\mathbf{z}(t_g + (q+1)\tau_\ell) &= \mathbf{z}(t_g + q\tau_\ell) - \int_{t_g+q\tau_\ell}^{t_g+(q+1)\tau_\ell} \eta_\ell(s)\hat{u}_{\ell,z}(s)\mathrm{d}s.
\end{aligned} \tag{A.16}$$

Using the decomposition $\mathcal{E}(t) = \text{term I} + \hat{\mathcal{E}}(t)$, one can express, and subsequently bound the sampling error as

$$\begin{aligned}
\hat{\mathcal{E}}(t) &= \left\langle \nabla f(\bar{\mathbf{x}}(t)), \frac{\eta_\ell(t)}{N}\mathbb{1}^T\cdot(u_{\ell,x}(t) - \hat{u}_{\ell,x}(t)) \right\rangle + \langle (I-R)\cdot\mathbf{y}(t), \eta_g(t)\cdot(u_g(t) - \hat{u}_g(t))\rangle\\
&\quad + \langle (I-R)\cdot\mathbf{y}(t), \eta_\ell(t)\cdot(u_{\ell,y}(t) - \hat{u}_{\ell,y}(t))\rangle\\
&\overset{(A.1)}{\le} \frac{\gamma_1(t)}{2}\|\nabla f(\bar{\mathbf{x}}(t))\|^2 + \frac{\gamma_2(t)}{2}\|(I-R)\cdot\mathbf{y}(t)\|^2\\
&\quad + \frac{\eta_g^2(t)}{2\gamma_2(t)}\|(I-R)\cdot(u_g(t) - \hat{u}_g(t))\|^2 + \frac{\eta_\ell^2(t)}{2\min\{N\gamma_1(t),\gamma_2(t)\}}\|u_{\ell,y}(t) - \hat{u}_{\ell,y}(t)\|^2\\
&\overset{(i)}{\le} \frac{\gamma_1(t)}{2}\|\nabla f(\bar{\mathbf{x}}(t))\|^2 + \frac{\gamma_2(t)}{2}\|(I-R)\cdot\mathbf{y}(t)\|^2 + \frac{\eta_g^2(t)}{2\gamma_2(t)}\|(I-R)\cdot(\mathbf{y}(t) - \mathbf{y}(t_g))\|^2\\
&\quad + \frac{L^2\eta_\ell^2(t)}{2\min\{N\gamma_1(t),\gamma_2(t)\}}\left(\|\mathbf{y}(t) - \mathbf{y}(t_\ell)\|^2 + \|\mathbf{z}(t) - \mathbf{z}(t_\ell)\|^2\right),
\end{aligned} \tag{A.17}$$

where in (i) we apply property P2 and (2.1) to the third term, such that $\|(I-R)\cdot(u_g(t) - \hat{u}_g(t))\|^2 = \|(I-R)\cdot W_A(\mathbf{y}(t) - \mathbf{y}(t_g))\|^2 \le \|(I-R)\cdot(\mathbf{y}(t) - \mathbf{y}(t_g))\|^2$, and we have used property P3 to the last term. The key is to bound the last three terms of (A.17). We divide it into three steps.

Step 1. We bound the third term involving $\|(I-R)\cdot(\mathbf{y}(t)-\mathbf{y}(t_g))\|^2$. With (A.2), we have $\|(I-R)\cdot(\mathbf{y}(t)-\mathbf{y}(t_g))\|^2 \le \|\mathbf{y}(t)-\mathbf{y}(t_g)\|^2$; then we bound the RHS by

$$
\begin{aligned}
\|\mathbf{y}(t)-\mathbf{y}(t_g)\|^2 &\overset{\text{(i)}}{=} \left\|(I-R)\cdot\int_{\tau_g}^{t}\eta_g(s)\hat{u}_g(s)\mathrm{d}s+\int_{t_g}^{t}\eta_\ell(s)\hat{u}_{\ell,y}(s)\mathrm{d}s\right\|^2\\
&\overset{\text{(ii)}}{\le} 2\tau_g^2\eta_g^2(t)\left\|\hat{u}_g(t)\right\|^2+2\left\|\int_{t_g}^{t}\eta_\ell(s)\hat{u}_{\ell,y}(s)\mathrm{d}s\right\|^2\\
&\overset{\text{(iii)}}{\le} 4\tau_g^2\eta_g^2(t)\left(\left\|\hat{u}_g(t)-u_g(t)\right\|^2+\left\|u_g(t)\right\|^2\right)+2\tau_\ell^2\sum_{\tau=t_g}^{t_\ell}\eta_\ell^2(\tau)\left\|\hat{u}_{\ell,y}(\tau)\right\|^2\\
&\overset{\text{(iv)}}{\le} 4\tau_g^2\eta_g^2(t)\left(\left\|\mathbf{y}(t)-\mathbf{y}(t_g)\right\|^2+\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2\right)+2\tau_\ell^2\sum_{\tau=t_g}^{t_\ell}\eta_\ell^2(\tau)\left\|\hat{u}_{\ell,y}(\tau)\right\|^2\\
&\overset{\text{(v)}}{\le} \frac{4\tau_g^2\eta_g^2(t)}{1-4\tau_g^2\eta_g^2(t)}\left\|(I-R)\cdot\mathbf{y}(t)\right\|^2+\frac{2\tau_\ell^2}{1-4\tau_g^2\eta_g^2(t)}\sum_{\tau=t_g}^{t_\ell}\eta_\ell^2(\tau)\left\|\hat{u}_{\ell,y}(\tau)\right\|^2, \qquad \text{(A.18)}
\end{aligned}
$$

where (i) uses the first relation in (A.16), and $R\cdot\hat{u}_g(t)=0$ (see property P1); in (ii) we apply Cauchy–Schwarz inequality and use the fact that $t-t_\ell\le\tau_g$ and $\hat{u}_g(s),\eta_g(s)$ remain constants in the integration; in (iii) we add and subtract $u_g(t)$ in the first term and applied Cauchy–Schwarz inequality, and (A.2); in (iv) we apply property P2 to the first term and get $\hat{u}_g(t)-u_g(t)=G_g(\mathbf{y}(t)-\mathbf{y}(t_g);A)$, and apply the second inequality in (2.1), and the last inequality in (A.2); (v) holds because we moved $\|\mathbf{y}(t)-\mathbf{y}(t_g)\|^2$ to the left and divide both sides by $1-4\tau_g^2\eta_g^2(t)$, and choose $\tau_g<\frac{1}{2\eta_g(t)}$ such that $4\tau_g^2\eta_g^2(t)<1$. To bound the last term of (A.18), we note that following series of relations:

$$
\begin{aligned}
\left\|\hat{u}_{\ell,y}(\tau)\right\|^2 &\le \left\|\hat{u}_\ell(\tau)\right\|^2 \le 2\left\|\hat{u}_\ell(\tau)-u_\ell(\tau)\right\|^2+2\left\|u_\ell(\tau)\right\|^2\\
&\overset{(P3)}{\le} 2L^2\cdot\left(\left\|\mathbf{y}(\tau)-\mathbf{y}(t_\ell)\right\|^2+\left\|\mathbf{z}(\tau)-\mathbf{z}(t_\ell)\right\|^2\right)+2\left\|u_\ell(\tau)\right\|^2\\
&\overset{(P4)}{\le} 2L^2\cdot\left(\left\|\mathbf{y}(\tau)-\mathbf{y}(t_\ell)\right\|^2+\left\|\mathbf{z}(\tau)-\mathbf{z}(t_\ell)\right\|^2\right)+2C_f\left\|\nabla f(\mathbf{x}(\tau))\right\|^2\\
&\le 2L^2\cdot\left(\left\|\mathbf{y}(\tau)-\mathbf{y}(t_\ell)\right\|^2+\left\|\mathbf{z}(\tau)-\mathbf{z}(t_\ell)\right\|^2\right)\\
&\quad+4C_f\left(\left\|\nabla f(\mathbf{x}(\tau))-\nabla f(\bar{\mathbf{x}}(\tau))\right\|^2+\left\|\nabla f(\bar{\mathbf{x}}(\tau))\right\|^2\right)\\
&\overset{(A2)}{\le} 2L^2\cdot\left(\left\|\mathbf{y}(\tau)-\mathbf{y}(t_\ell)\right\|^2+\left\|\mathbf{z}(\tau)-\mathbf{z}(t_\ell)\right\|^2\right)\\
&\quad+4C_f\left(L_f^2\left\|(I-R)\cdot\mathbf{x}(\tau))\right\|^2+\left\|\nabla f(\bar{\mathbf{x}}(\tau))\right\|^2\right),
\end{aligned}
\qquad \text{(A.19)}
$$

where C_f is defined in (A.15). Note that we need to further bound $\|\mathbf{y}(\tau)-\mathbf{y}(t_\ell)\|^2+\|\mathbf{z}(\tau)-\mathbf{z}(t_\ell)\|^2$, which is the same as the last two terms in (A.16).

Step 2. We then bound $\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\|^2+\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\|^2$. By (A.16), we have

$$
\begin{aligned}
\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\|^2+\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\|^2 &\overset{\text{(A.19)}}{=} \left\|\int_{t_\ell}^{t}\eta_g(s)\hat{u}_g(s)+\eta_\ell(s)\cdot\hat{u}_\ell(s)\mathrm{d}s\right\|^2\\
&\overset{\text{(i)}}{\le} 2\tau_\ell^2\eta_g^2(t)\left\|\hat{u}_g(t)\right\|^2+2\tau_\ell^2\eta_\ell^2(t)\cdot\left\|\hat{u}_\ell(t)\right\|^2\\
&\overset{\text{(A.19)}}{\le} 2\tau_\ell^2\eta_g^2(t)\left\|\hat{u}_g(t)\right\|^2+4L^2\tau_\ell^2\eta_\ell^2(t)\cdot\left(\left\|\mathbf{y}(t)-\mathbf{y}(t_\ell)\right\|^2+\left\|\mathbf{z}(t)-\mathbf{z}(t_\ell)\right\|^2\right)
\end{aligned}
\qquad \text{(A.20)}
$$

$$
\begin{aligned}
&+ 8L^2 C_f \tau_\ell^2 \eta_\ell^2(t) \cdot \left(\|\nabla f(\bar{\mathbf{x}}(t))\|^2 + L_f^2 \|(I-R)\cdot \mathbf{x}(t)\|^2 \right) \\
&\overset{\text{(ii)}}{\leq} \frac{4\tau_\ell^2 \eta_g^2(t)}{1 - 4L^2\tau_\ell^2\eta_\ell^2(t)} \left(\|u_g(t) - \hat{u}_g(t)\|^2 + \|u_g(t)\|^2 \right) \\
&\quad + \frac{8L^2 C_f \tau_\ell^2 \eta_\ell^2(t)}{1 - 4L^2\tau_\ell^2\eta_\ell^2(t)} \cdot \left(\|\nabla f(\bar{\mathbf{x}}(t))\|^2 + L_f^2 \|(I-R)\cdot \mathbf{x}(t)\|^2 \right) \\
&\overset{\text{(iii)}}{\leq} \frac{4\tau_\ell^2 \eta_g^2(t)}{1 - 4L^2\tau_\ell^2\eta_\ell^2(t)} \left(\|\mathbf{y}(t) - \mathbf{y}(t_g)\|^2 + \|(I-R)\cdot \mathbf{y}(t)\|^2 \right) \\
&\quad + \frac{8L^2 C_f \tau_\ell^2 \eta_\ell^2(t)}{1 - 4L^2\tau_\ell^2\eta_\ell^2(t)} \cdot \left(\|\nabla f(\bar{\mathbf{x}}(t))\|^2 + L_f^2 \|(I-R)\cdot \mathbf{x}(t)\|^2 \right),
\end{aligned}
$$

where in (i) we apply Cauchy–Schwarz inequality; in (ii) add and subtract $u_g(t)$ to the first term and move $\|\mathbf{y}(t) - \mathbf{y}(t_\ell)\|^2 + \|\mathbf{z}(t) - \mathbf{z}(t_\ell)\|^2$ to the left and divide both sides by $1 - 4L^2\tau_\ell^2\eta_\ell^2(t)$, and choose $\tau_\ell < \frac{1}{2L\eta_\ell(t)}$ such that $4L^2\tau_\ell^2\eta_\ell^2(t) < 1$; in (iii) we apply the second inequality in (2.1), as well as the fact that $\|I - R\| \leq 1$.

To proceed, let us define $C_{43} := \frac{4\tau_g^2\eta_g^2(t)}{1-4\tau_g^2\eta_g^2(t)}$, $C_{44} := \frac{2\tau_\ell^2\eta_\ell^2(t)}{1-4\tau_g^2\eta_g^2(t)}$, $C_{45} := \frac{4\tau_\ell^2\eta_g^2(t)}{1-4L^2\tau_\ell^2\eta_\ell^2(t)}$, and $C_{46} := \frac{8L^2C_f\tau_\ell^2\eta_\ell^2(t)}{1-4L^2\tau_\ell^2\eta_\ell^2(t)}$. Then by plugging (A.18) into (A.20), we have

(A.21)

$$
\begin{aligned}
&\|\mathbf{y}(t) - \mathbf{y}(t_\ell)\|^2 + \|\mathbf{z}(t) - \mathbf{z}(t_\ell)\|^2 \overset{\text{(i)}}{\leq} \left(C_{45} + C_{43}C_{45} + C_{46}L_f^2\right) \cdot \|(I-R)\cdot \mathbf{y}(t)\|^2 \\
&\quad + C_{46} \|\nabla f(\bar{\mathbf{x}}(t))\|^2 + QC_{44}C_{45} \cdot \sum_{\tau=t_g}^{t_\ell} \|\hat{u}_{\ell,y}(\tau)\|^2 \\
&\overset{\text{(ii)}}{\leq} \left(C_{45} + C_{43}C_{45} + C_{46}L_f^2\right) \cdot \|(I-R)\cdot \mathbf{y}(t)\|^2 + C_{46} \|\nabla f(\bar{\mathbf{x}}(t))\|^2 \\
&\quad + QC_{44}C_{45} \cdot \sum_{\tau=t_g}^{t_\ell} (C_x^2 + C_v^2) \cdot \left(\|\nabla f(\mathbf{x}(\tau)) - \nabla f(\bar{\mathbf{x}}(\tau))\|^2 + \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \right) \\
&\overset{(A2)}{\leq} \left(C_{45} + C_{43}C_{45} + C_{46}L_f^2\right) \cdot \|(I-R)\cdot \mathbf{y}(t)\|^2 + C_{46} \|\nabla f(\bar{\mathbf{x}}(t))\|^2 \\
&\quad + QC_{44}C_{45} \cdot \sum_{\tau=t_g}^{t_\ell} (C_x^2 + C_v^2) \cdot \left(L_f^2 \|(I-R)\cdot \mathbf{x}(\tau)\|^2 + \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \right),
\end{aligned}
$$

where in (i) we use the fact that $t - t_g \leq Q\tau_\ell$; in (ii) we first apply property P4 to the last term, then subtract $\nabla f(\bar{\mathbf{x}}(\tau))$, and finally used Cauchy–Schwartz inequality. This completes part II of the proof.

Step 3. Finally, we substitute (A.21) into part I (A.19); then to (A.18), we obtain

$$
\begin{aligned}
\|\mathbf{y}(t) - \mathbf{y}(t_g)\|^2 &\overset{\text{(A.19)}}{\leq} 4C_f C_{44} \sum_{\tau=t_g}^{t} \left(L_f^2 \|(I-R)\cdot \mathbf{x}(\tau)\|^2 + \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \right) \\
&\quad + C_{43} \|(I-R)\cdot \mathbf{y}(t)\|^2 + 2L^2 C_{44} \sum_{\tau=t_g}^{t} \left(\|\mathbf{y}(\tau) - \mathbf{y}(t_\ell)\|^2 + \|\mathbf{z}(\tau) - \mathbf{z}(t_\ell)\|^2 \right) \\
&\overset{\text{(A.21)}}{\leq} C_{43} \|(I-R)\cdot \mathbf{y}(t)\|^2 + 4C_f C_{44} \sum_{\tau=t_g}^{t} \left(L_f^2 \|(I-R)\cdot \mathbf{x}(\tau)\|^2 + \|\nabla f(\bar{\mathbf{x}}(\tau))\|^2 \right)
\end{aligned}
$$

$$+ 2L^2 C_{44} \sum_{\tau=t_g}^{t} C_{46} \left\|\nabla f(\bar{\mathbf{x}}(\tau))\right\|^2$$
$$+ 2L^2 C_{44}^2 C_{45} \cdot \sum_{\tau=t_g}^{t} \sum_{\tau_1=t_g}^{\tau} (C_x^2 + C_v^2) \cdot \left(L_f^2 \left\|(I-R)\cdot \mathbf{x}(\tau_1)\right\|^2 + \left\|\nabla f(\bar{\mathbf{x}}(\tau_1))\right\|^2 \right).$$

Then we substitute (A.18) and (A.21) into (A.17); then to (A.4), we obtain

$$\int_0^t \dot{\mathcal{E}}(\tau)\mathrm{d}\tau \le -\int_0^t \left(\frac{\gamma_1(\tau)}{2} - C_{41}(\tau)\right) \cdot \left\|\nabla f(\bar{\mathbf{x}}(\tau))\right\|^2 \mathrm{d}\tau$$
$$- \int_0^t \left(\frac{\gamma_2(\tau)}{2} - C_{42}(\tau)\right) \cdot \left\|(I-R)\cdot \mathbf{y}(\tau)\right\|^2 \mathrm{d}\tau,$$

where we have defined

$$C_{41} := \frac{L^2 \eta_\ell^2(\tau) \cdot \left(C_{45} \cdot (1 + L_f^2 C_{47} + C_{45}) + C_{46} L_f^2 \right)}{2\min\{N\gamma_1(\tau), \gamma_2(\tau)\}} + \frac{C_g \eta_g^2(\tau) \cdot (C_{43} + L_f^2 C_{47})}{2\gamma_2(\tau)},$$
$$C_{42} := \frac{L^2 \eta_\ell^2(\tau) \cdot (C_{46} + C_{45} C_{47})}{2\min\{N\gamma_1(\tau), \gamma_2(\tau)\}} + \frac{C_g \eta_g^2(\tau) C_{47}}{2\gamma_2(\tau)} \text{ and } C_{47} := Q^2 C_{44}^2 \cdot (C_x^2 + C_v^2).$$

Appendix B. Distributed algorithms as discretized multirate systems. In this section, we provide additional discussions on how to map the distributed algorithms to the discretized multirate systems. First, let us discuss decentralized algorithms.

DGD [20]. The updates are given by (where $c > 0$ is the stepsize)

$$\mathbf{x}(k+1) = W\mathbf{x}(k) - c\nabla f(\mathbf{x}(k)) = \mathbf{x}(k) - ((I-W)\mathbf{x}(k) + c) \cdot \nabla f(\mathbf{x}(k)).$$

It uses the discretization case III, with the following continuous-time controllers:

$$u_{g,x} = (I-W)\cdot \mathbf{x}, \quad u_{\ell,x} = \nabla f(\mathbf{x}).$$

DLM [17]. The updates are given by

$$\mathbf{x}(k+1) = \mathbf{x}(k) - \eta \cdot (\nabla f(\mathbf{x}(k)) + c \cdot (I-W)\cdot \mathbf{x}(k) + \mathbf{v}(k)),$$
$$\mathbf{v}(k+1) = \mathbf{v}(k) + c \cdot (I-W)\cdot \mathbf{x}(k+1).$$

It corresponds to case III, with the following continuous-time controllers:

$$u_{g,x} = c\cdot(I-W)\cdot \mathbf{x} + \mathbf{v}, \quad u_{g,v} = (I-W)\cdot \mathbf{x}, \quad u_{\ell,x} = \nabla f(\mathbf{x}), \quad u_{\ell,v} = 0.$$

Next, we discuss some popular FL algorithms. For this class of algorithms, the agents are connected with a central server which performs averaging. The corresponding communication graph is a fully connected graph, with the weight matrix being the averaging matrix, i.e., $W = R, W_A = I - R$.

FedProx [14]. The updates are given by (where GD is used to solve local problems):

$$\mathbf{x}(k+1) = \begin{cases} \mathbf{x}(k) - \eta_1 \nabla f(\mathbf{x}(k)) - \eta_2(\mathbf{x}(k) - \mathbf{x}(k_0)), \ k \bmod Q \neq 0, \ k_0 = k - (k \bmod Q), \\ R\mathbf{x}(k) - \eta_1 \nabla f(\mathbf{x}(k)) - \eta_2 \cdot (\mathbf{x}(k) - \mathbf{x}(k_0)), k \bmod Q = 0. \end{cases}$$

It uses the discretization case I, with the following continuous-time controllers:

$$u_{g,x} = (I-R)\cdot \mathbf{x}, \quad u_{\ell,x} = \nabla f(\mathbf{x}).$$

FedPD [36]. The updates are given by (where GD is used to solve local problems):

$$\mathbf{x}(k+1) = \mathbf{x}(k) - \eta_1 \cdot (\nabla f(\mathbf{x}(k)) + \mathbf{v}(k) + \eta_2 \cdot (\mathbf{x}(k_0) - R\mathbf{x}(k_0))),\ k_0 = k - (k \bmod Q),$$
$$\mathbf{w}(k+1) = \begin{cases} R\mathbf{x}(k), k \bmod Q = 0, \\ \mathbf{w}(k), k \bmod Q \neq 0, \end{cases}$$
$$\mathbf{v}(k+1) = \begin{cases} \mathbf{v}(k) + \frac{1}{\eta_2} \cdot (\mathbf{x}(k) - \mathbf{w}(k)), k \bmod Q = 0, \\ \mathbf{v}(k), k \bmod Q \neq 0. \end{cases}$$

It uses the discretization cases I or IV. Observe that $\mathbf{w}$ tracks $R\mathbf{x}$. Replace $\mathbf{w}$ with $R\mathbf{x}$, we can obtain the following controller:

$$u_{g,x} = (I - R) \cdot \mathbf{x} + \mathbf{v}, \quad u_{g,v} = -(I - R) \cdot \mathbf{x}, \quad u_{\ell,x} = \nabla f(\mathbf{x}), \quad u_{\ell,v} = 0.$$

Finally, we discuss one more rate optimal algorithm.

D-GPDA [27]. The update step of the Distributed Gradient Primal-Dual Algorithm (D-GPDA) is given by

$$\begin{aligned} \mathbf{x}(k+1) = \arg\min_{\mathbf{x}} & \left\langle \nabla f(\mathbf{x}(k)) + A^T \mathbf{v}(k), \mathbf{x} - \mathbf{x}(k) \right\rangle \\ & + \frac{1}{2} \left\| \eta_1 A \mathbf{x} \right\|^2 + \left\| \eta_1 |A| \cdot (\mathbf{x} - \mathbf{x}(k)) \right\|^2 + \left\| \eta_2 \cdot (\mathbf{x} - \mathbf{x}(k)) \right\|^2 \\ \mathbf{v}(k+1) = & \mathbf{v}(k) + \eta_1^2 A \mathbf{x}(k+1), \end{aligned}$$

where $\mathbf{v}$ is the dual variable for the linear consensus constraint. By assuming the minimization is solved with gradient flow or K-step gradient descent, this algorithm is using the discretization case II, with the following continuous-time controllers:

$$u_{g,x} = \eta_1 W \mathbf{x} + \eta_2 \cdot (\mathbf{x} - \mathbf{v}_2) - \eta_1 \left| A^T A \right| \mathbf{v}_2 + A^T \mathbf{v}_1, \quad u_{g,v} = [-\eta_1^2 A \mathbf{x};\ 0],$$
$$u_{\ell,x} = \nabla f(\mathbf{x}), \quad u_{\ell,v} = [0;\ -(\mathbf{x} - \mathbf{v}_2)].$$

REFERENCES

[1] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečn, S. Mazzocchi, B. McMahan, T. Van Overveldt, D. Petrou, D. Ramage, and J. Roselander, *Towards federated learning at scale: System design*, Proc. Mach. Learn. Syst., 1 (2019), pp. 374–388.

[2] S. Bubeck, Y. T. Lee, and M. Singh, *A Geometric Alternative to Nesterov's Accelerated Gradient Descent*, preprint, https://arxiv.org/abs/1506.08187, 2015.

[3] T.-H. Chang, M. Hong, H.-T. Wai, X. Zhang, and S. Lu, *Distributed learning in the nonconvex world: From batch data to streaming and beyond*, IEEE Signal Proc. Mag., 37 (2020), pp. 26–38.

[4] P. Di Lorenzo and G. Scutari, *Next: In-network nonconvex optimization*, IEEE Trans. Signal Inf. Process. Netw., 2 (2016), pp. 120–136.

[5] G. Droge, H. Kawashima, and M. B. Egerstedt, *Continuous-time proportional-integral distributed optimisation for networked systems*, J. Control Decis., 1 (2014), pp. 191–213.

[6] G. França, D. P. Robinson, and R. Vidal, *A Nonsmooth Dynamical Systems Perspective on Accelerated Extensions of ADMM*, preprint, https://arxiv.org/abs/1808.04048, 2018.

[7] E. Ghadimi, M. Johansson, and I. Shames, *Accelerated gradient methods for networked optimization*, in Proceedings of the 2011 American Control Conference, IEEE, 2011, pp. 1668–1673.

[8] B. Hu and L. Lessard, *Control interpretations for first-order optimization methods*, in Proceedings of the 2017 American Control Conference (ACC), IEEE, 2017, pp. 3114–3119.

[9] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, *Scaffold: Stochastic controlled averaging for federated learning*, in Proceedings of the International Conference on Machine Learning, PMLR, 2020, pp. 5132–5143.

[10] A. Khaled, K. Mishchenko, and P. Richtárik, *First Analysis of Local GD on Heterogeneous Data*, preprint, https://arxiv.org/abs/1909.04715, 2019.
[11] B. C. Kuo, *Digital Control Systems*, Holt, Rinehart and Winston, New York, 1980.
[12] L. Lessard, B. Recht, and A. Packard, *Analysis and design of optimization algorithms via integral quadratic constraints*, SIAM J. Optim., 26 (2016), pp. 57–95, https://doi.org/10.1137/15M1009597.
[13] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, *Federated learning: Challenges, methods, and future directions*, IEEE Signal Proc. Mag., 37 (2020), pp. 50–60.
[14] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, *Federated optimization in heterogeneous networks*, Proc. Mach. Learn. Syst., 2 (2020), pp. 429–450.
[15] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, *On the convergence of fedavg on non-iid data*, in Proceedings of the International Conference on Learning Representations, 2020.
[16] Z. Li, W. Shi, and M. Yan, *A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates*, IEEE Trans. Signal Process., 67 (2019), pp. 4494–4506.
[17] Q. Ling, W. Shi, G. Wu, and A. Ribeiro, *DDLM Decentralized linearized alternating direction method of multipliers*, IEEE Trans. Signal Process., 63 (2015), pp. 4051–4064.
[18] S. Lu, X. Zhang, H. Sun, and M. Hong, *NSD: A gradient-tracking based nonconvex stochastic algorithm for decentralized optimization*, in Proceedings of the 2019 IEEE Data Science Workshop (DSW), IEEE, 2019, pp. 315–321.
[19] M. Muehlebach and M. Jordan, *A dynamical systems perspective on nesterov acceleration*, in Proceedings of the International Conference on Machine Learning, 2019, pp. 4656–4662.
[20] A. Nedic and A. Ozdaglar, *Distributed subgradient methods for multi-agent optimization*, IEEE Trans. Automat. Control, 54 (2009), pp. 48–61.
[21] A. Olshevsky and J. N. Tsitsiklis, *Convergence speed in distributed consensus and averaging*, SIAM J. Control Optim., 48 (2009), pp. 33–55, https://doi.org/10.1137/060678324.
[22] A. Orvieto and A. Lucchi, *Continuous-time models for stochastic optimization algorithms*, Adv. Neural Inf. Process., 32 (2019), pp. 12610–12622.
[23] R. A. Poliquin and R. T. Rockafellar, *Generalized hessian properties of regularized nonsmooth functions*, SIAM J. Optim., 6 (1996), pp. 1121–1137, https://doi.org/10.1137/S1052623494279316.
[24] S. J. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečn, S. Kumar, and H. B. McMahan, *Adaptive federated optimization*, in Proceedings of the International Conference on Learning Representations, 2020.
[25] R. Rossi and G. Savaré, *Gradient flows of non convex functionals in Hilbert spaces and applications*, ESAIM: Control Optim. Calc. Var., 12 (2006), pp. 564–614.
[26] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, *Optimal algorithms for smooth and strongly convex distributed optimization in networks*, in Proceedings of the International Conference on Machine Learning, PMLR, 2017, pp. 3027–3036.
[27] H. Sun and M. Hong, *Distributed non-convex first-order optimization and information processing: Lower complexity bounds and rate optimal algorithms*, IEEE Trans. Signal Process., 67 (2019), pp. 5912–5928.
[28] H. Sun, S. Lu, and M. Hong, *Improving the sample and communication complexity for decentralized non-convex optimization: Joint gradient estimation and tracking*, in Proceedings of the International Conference on Machine Learning, PMLR, 2020, pp. 9217–9228.
[29] A. Sundararajan, *Analysis and Design of Distributed Optimization Algorithms*, The University of Wisconsin-Madison, Madison, WI, 2021.
[30] B. Swenson, R. Murray, H. V. Poor, and S. Kar, *Distributed gradient descent: Nonconvergence to saddle points and the stable-manifold theorem*, in Proceedings of the 2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton), IEEE, 2019, pp. 595–601.
[31] B. Swenson, R. Murray, H. V. Poor, and S. Kar, *Distributed gradient flow: Nonsmoothness, nonconvexity, and saddle point evasion*, IEEE Trans. Automat. Control, 67 (2022), pp. 3949–3964.
[32] J. Wang and N. Elia, *A control perspective for centralized and distributed convex optimization*, in Proceedings of the 2011 50th IEEE Conference on Decision and Control and European Control Conference, IEEE, 2011, pp. 3800–3805.
[33] H. Ye, L. Luo, Z. Zhou, and T. Zhang, *Multi-Consensus Decentralized Accelerated Gradient Descent*, preprint, https://arxiv.org/abs/2005.00797, 2020.
[34] K. Yuan, Q. Ling, and W. Yin, *On the convergence of decentralized gradient descent*, SIAM J. Optim., 26 (2016), pp. 1835–1854.

[35] K. Yuan, W. Xu, and Q. Ling, *Can primal methods outperform primal-dual methods in decentralized dynamic optimization?*, IEEE Trans. Signal Process., 68 (2020), pp. 4466–4480.

[36] X. Zhang, M. Hong, S. Dhople, W. Yin, and Y. Liu, *FedPD: A federated learning framework with adaptivity to non-iid data*, IEEE Trans. Signal Process., 69 (2021), pp. 6055–6070.

[37] X. Zhang, M. Hong, and N. Elia, *Understanding a Class of Decentralized and Federated Optimization Algorithms: A Multi-Rate Feedback Control Perspective*. Available online at http://people.ece.umn.edu/~mhong/unified.pdf.