Periodic and event-triggered communication for distributed continuous-time convex optimization

Solmaz S. Kia Jorge Cortés Sonia Martínez

***Abstract*— We propose a distributed continuous-time algorithm to solve a network optimization problem where the global cost function is a strictly convex function composed of the sum of the local cost functions of the agents. We establish that our algorithm, when implemented over strongly connected and weight-balanced directed graph topologies, converges exponentially fast when the local cost functions are strongly convex and their gradients are globally Lipschitz. We also characterize the privacy preservation properties of our algorithm and extend the convergence guarantees to the case of time-varying, strongly connected, weight-balanced digraphs. When the network topology is a connected undirected graph, we show that exponential convergence is still preserved if the gradients of the strongly convex local cost functions are locally Lipschitz, while it is asymptotic if the local cost functions are convex. We also study discrete-time communication implementations. Specifically, we provide an upper bound on the stepsize of a synchronous periodic communication scheme that guarantees convergence over connected undirected graph topologies and, building on this result, design a centralized event-triggered implementation that is free of Zeno behavior. Simulations illustrate our results.**

I. Introduction

Distributed optimization problems are pervasive in many scenarios, including parallel systems, distributed computation, and multi-agent systems [1], [2], [3]. A common class of distributed convex optimization problems considers the constrained or unconstrained optimization of a sum of local convex functions, which represent private local costs only available to each agent. Such problems model a wide range of practical network operations where the global cost function is a performance metric consisting of a sum of local private utility functions. Most of the current distributed optimization solvers for these problems are discrete-time algorithms [4], [5], [6], [7], [8] which employ consensus-based dynamics to arrive at the solution. More recently, a number of continuous-time dynamical solvers [9], [10], [11], [12] have been introduced whose convergence properties are studied using control-theoretic tools. Taking this perspective on the design and analysis of optimization algorithms facilitates the characterization of properties such as speed of convergence, disturbance rejection, and robustness to parameter and model uncertainties. This manuscript further contributes to this body of work. Motivated by the practical constraints on communication imposed by real-time implementations, we also explore the development of distributed convex optimization strategies that have agents performing computation in continuous time but only require communication between neighbors at discrete instants of time. We study the stability and convergence properties of our proposed algorithm through standard Lyapunov analysis.

The authors are with the Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA 92093, USA `{skia,cortes,soniamd}@ucsd.edu`

Literature review of continuous-time optimization algorithms: The continuous-time, unconstrained convex optimization algorithms proposed in [11], [12] are second-order algorithms which use the inverse of the Hessian. These algorithms require a special initialization and are guaranteed to converge only for connected undirected graph topologies. The algorithm proposed in [9] is a gradient-based scheme whose convergence guarantees are valid for connected undirected graph topologies. A variation of this algorithm with additional convergence properties over strongly connected and weight-balanced digraph topologies is presented in [10]. The protocol of [10] is obtained by introducing a gain in the algorithm of [9] and characterizing the admissible range for which this gain ensures convergence. The convergence of both algorithms in [9], [10] is asymptotic. In general, the works mentioned above do not characterize the privacy preservation properties of the proposed algorithms. Privacy preservation is a crucial requirement in network applications, see e.g., [13] and references therein. In our specific setup, privacy preservation is concerned with determining whether agents inside or outside the network can discover any information about the local cost functions by listening to the communication messages.

Statement of contributions: We consider an unconstrained convex optimization problem whose objective function is strictly convex and can be written as a sum of local cost functions, one per agent. We propose a novel gradient-based distributed algorithm for networks with strongly connected and weight-balanced digraph topologies. We show that the algorithm has an exponential rate of convergence when the local cost functions are m-strongly convex and their gradients are globally Lipschitz and characterize its privacy preservation properties. The results are also valid for networks with time-varying interaction topologies as long as the digraph stays strongly connected and weight-balanced. For connected undirected graph topologies, we show that the global Lipschitzness of the local gradients can be relaxed to local Lipschitzness. Also, for this case, we prove that the algorithm converges, asymptotically, when the local cost functions are convex. For implementation of the algorithm over networks with wireless communication, we also study discrete-time and event-triggered communication implementations of the proposed algorithm. For networks with connected graph topologies, we obtain an upper bound on the suitable stepsizes that guarantee convergence for a periodic discrete-time communication implementation. We build on this result to design a centralized event-triggered communication implementation which is free of Zeno behavior.

Organization: Section II introduces basic notation and concepts from graph theory and convex functions. Section III presents the problem statement. Section IV introduces our novel continuous-time distributed convex optimization algorithm and characterizes its properties on convergence and privacy preservation. Section V discusses continuous-time implementations with discrete-time-communication of the proposed algorithm. Section VI illustrates our results in simulation. Finally, Section VII gathers our conclusions and ideas for future work. Due to the space limitations, the proofs are omitted and will appear elsewhere.

II. Preliminaries

In this section, we introduce our notation and some basic concepts from convex functions and graph theory.

A. Notation

Let $\mathbb{R}$ and $\mathbb{N}$ denote, respectively, the set of real and natural numbers. We use $\Re(\cdot)$ to represent the real part of a complex number. The transpose of a matrix $\mathbf{A}$ is $\mathbf{A}^\top$. We let $\mathbf{1}_n$ (resp. $\mathbf{0}_n$) denote the vector of n ones (resp. n zeros), and denote by $\mathbf{I}_n$ the $n\times n$ identity matrix. We let $\mathbf{\Pi}_n = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top$. When clear from the context, we do not specify the matrix dimensions. For $\mathbf{A}\in\mathbb{R}^{n\times m}$ and $\mathbf{B}\in\mathbb{R}^{p\times q}$, we let $\mathbf{A}\otimes\mathbf{B}$ denote their Kronecker product. For $\mathbf{u}\in\mathbb{R}^d$, $\|\mathbf{u}\| = \sqrt{\mathbf{u}^\top\mathbf{u}}$ denotes the standard Euclidean norm. For vectors $\mathbf{u}_1,\cdots,\mathbf{u}_m$, we let $\mathbf{u}=(\mathbf{u}_1,\cdots,\mathbf{u}_m)$ represent the aggregated vector. In a networked system, we distinguish the local variables at each agent by a superscript, e.g., $\mathbf{x}^i$ is the local state of agent i. If $\mathbf{p}^i\in\mathbb{R}^d$ is a variable of agent i, the aggregated $\mathbf{p}^i$'s of the network of N agents is represented by $\mathbf{p}=(\mathbf{p}^1,\cdots,\mathbf{p}^N)\in(\mathbb{R}^d)^N$. A differentiable function $f:\mathbb{R}^d\to\mathbb{R}$ is *strictly convex* over a convex set $C\subset\mathbb{R}^d$ iff

$$(\mathbf{z}-\mathbf{x})^\top(\nabla f(\mathbf{z})-\nabla f(\mathbf{x}))>0,\quad \forall\, \mathbf{x},\mathbf{z}\in C,\ \mathbf{x}\neq\mathbf{z},$$

and it is m*-strongly convex* ($m>0$) iff

$$(\mathbf{z}-\mathbf{x})^\top(\nabla f(\mathbf{z})-\nabla f(\mathbf{x}))\geq m\|\mathbf{z}-\mathbf{x}\|^2,\ \forall\,\mathbf{x},\mathbf{z}\in C,\ \mathbf{x}\neq\mathbf{z}.$$

A function $\mathbf{f}:\mathbb{R}^d\to\mathbb{R}^d$ is Lipschitz with constant $M>0$, or simply M*-Lipschitz*, over a set $C\subset\mathbb{R}^d$ iff

$$\|\mathbf{f}(\mathbf{x})-\mathbf{f}(\mathbf{y})\|\leq M\,\|\mathbf{x}-\mathbf{y}\|\,,\quad \forall \mathbf{x},\mathbf{y}\in C.$$

B. Graph Theory

Here, we briefly review some basic concepts from graph theory and linear algebra following [14]. A *directed graph*, or simply a *digraph*, is a pair $\mathcal{G}=(\mathcal{V},\mathcal{E})$, where $\mathcal{V}=\{1,\dots,N\}$ is the *node set* and $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$ is the *edge set*. An edge from i to j, denoted by (i,j), means that agent j can send information to agent i. For an edge $(i,j)\in\mathcal{E}$, i is called an *in-neighbor* of j and j is called an *out-neighbor* of i. A graph is *undirected* if $(i,j)\in\mathcal{E}$ anytime $(j,i)\in\mathcal{E}$. A *directed path* is a sequence of nodes connected by edges.

A *weighted digraph* is a triplet $\mathcal{G}=(\mathcal{V},\mathcal{E},\mathsf{A})$, where $(\mathcal{V},\mathcal{E})$ is a digraph and $\mathsf{A}\in\mathbb{R}^{N\times N}$ is a weighted *adjacency* matrix with the property that $\mathsf{a}_{\mathsf{ij}}>0$ if $(i,j)\in\mathcal{E}$ and $\mathsf{a}_{\mathsf{ij}}=0$, otherwise. A weighted digraph is *undirected* if $\mathsf{a}_{\mathsf{ij}}=\mathsf{a}_{\mathsf{ji}}$ for all $i,j\in\mathcal{V}$. We refer to a strongly connected and undirected graph as a *connected graph*. The *weighted out-degree* and *weighted in-degree* of a node i, are respectively, $\mathsf{d}_{\text{in}}^i=\sum_{j=1}^N a_{ji}$ and $\mathsf{d}_{\text{out}}^i=\sum_{j=1}^N \mathsf{a}_{ij}$. A digraph is *weight-balanced* if at each node $i\in\mathcal{V}$, the weighted out-degree and weighted in-degree coincide (although they might be different across different nodes). The *(out-) Laplacian* matrix is $\mathsf{L}=\mathsf{D}^{\text{out}}-\mathsf{A}$, where $\mathsf{D}^{\text{out}}=\text{Diag}(\mathsf{d}_{\text{out}}^1,\cdots,\mathsf{d}_{\text{out}}^N)\in\mathbb{R}^{N\times N}$. Note that $\mathsf{L}\mathbf{1}_N=\mathbf{0}$. A digraph is weight-balanced if and only if $\mathbf{1}_N^T\mathsf{L}=\mathbf{0}$ if and only if $\text{Sym}(\mathsf{L})=\frac{1}{2}(\mathsf{L}+\mathsf{L}^T)$ is positive semi-definite. Based on the structure of L, at least one of the eigenvalues of L is zero and the rest of them have nonnegative real parts. We denote the eigenvalues of L by $\lambda_1,\dots,\lambda_N$, where $\lambda_1=0$ and $\Re(\lambda_i)\leq\Re(\lambda_j)$, for $i<j$, and the eigenvalues of $\text{Sym}(\mathsf{L})$ by $\hat{\lambda}_1,\dots,\hat{\lambda}_N$. For a strongly connected and weight-balanced digraph, zero is a simple eigenvalue of both $\mathbf{L}$ and $\text{Sym}(\mathsf{L})$. In this case, we order the eigenvalues of $\text{Sym}(\mathsf{L})$ as $\hat{\lambda}_1=0<\hat{\lambda}_2\leq\hat{\lambda}_3\leq\cdots\leq\hat{\lambda}_N$. For convenience, we define $\mathbf{L}=\mathsf{L}\otimes\mathsf{I}_d$ and $\mathbf{\Pi}=\mathsf{\Pi}_N\otimes\mathsf{I}_d$ to deal with variables of dimension $d\in\mathbb{N}$.

III. Problem Definition

Consider a network of N agents with interaction topology described by a strongly connected, weight-balanced digraph $\mathcal{G}$. Each agent $i\in\{1,\dots,N\}$ is endowed with a local cost function $f^i:\mathbb{R}^d\to\mathbb{R}$ which is assumed differentiable. The global network cost function $f:\mathbb{R}^d\to\mathbb{R}$ is defined as $f(\mathsf{x})=\Sigma_{i=1}^N f^i(\mathsf{x})$. We assume this function to be strictly convex. Our objective is to design a distributed optimization algorithm such that each agent obtains the global minimizer $-\infty<\mathsf{x}^\star<\infty$ of the feasible optimization problem

$$\mathsf{x}^\star=\arg\min_{\mathsf{x}\in\mathbb{R}^d} f(\mathsf{x})$$

using only its own local data and exchanged information with its neighbors (note that the strict convexity of f implies the uniqueness of the optimizer). We are also interested in characterizing the privacy preservation properties of the algorithmic solution to this distributed optimization problem. Specifically, we aim to identify conditions guaranteeing that no information about the local cost function of an agent is revealed to, or can be reconstructed by, any other agent in the network.

IV. Distributed Solution for Convex Optimization

To solve the distributed optimization problem of Section III, we propose the following distributed optimization algorithm

$$\dot{\mathbf{v}}^i = \alpha\beta \sum_{j=1}^{N} \mathsf{a}_{ij}(\mathbf{x}^i - \mathbf{x}^j), \tag{1a}$$

$$\dot{\mathbf{x}}^i = -\alpha\nabla f^i(\mathbf{x}^i) - \beta \sum_{j=1}^{N} \mathsf{a}_{ij}(\mathbf{x}^i - \mathbf{x}^i) - \mathbf{v}^i, \tag{1b}$$

for $i \in \{1, \ldots, N\}$, with $\alpha > 0$, $\beta > 0$. The collective form of this algorithm is as follows

$$\dot{\mathbf{v}} = \alpha\beta\mathbf{L}\mathbf{x}, \tag{2a}$$

$$\dot{\mathbf{x}} = -\alpha\nabla\tilde{f}(\mathbf{x}) - \beta\mathbf{L}\mathbf{x} - \mathbf{v}. \tag{2b}$$

Here, $\tilde{f} : (\mathbb{R}^d)^N \to \mathbb{R}$ is defined by $\tilde{f}(\mathbf{x}) = \sum_{i=1}^{N} f^i(\mathbf{x}^i)$. This algorithm is distributed because each agent only needs to receive information from its out-neighbors about their corresponding variables in $\mathbf{x}$. In contrast, the continuous-time coordination algorithms in [9], [10] require the communication of the corresponding variables in both $\mathbf{x}$ and $\mathbf{v}$. In the following, we study the stability and convergence properties of the algorithm (1) over directed and undirected graphs.

A. Strongly Connected, Weight-Balanced Digraphs

Here, we study the convergence of the distributed optimization algorithm (1) over strongly connected and weight-balanced digraph topologies. We first consider the case where the interaction topology is fixed, and then discuss time-varying interaction topologies. The following result identifies conditions on the local cost functions $\{f^i\}_{i=1}^N$ and the parameter β to guarantee the exponential convergence of (1) to the solution of the distributed optimization problem.

Theorem 4.1 (Convergence of (1) over strongly connected and weight-balanced digraphs): Let $\mathcal{G}$ be a strongly connected and weight-balanced digraph. Assume the local cost function f^i, $i \in \{1, \ldots, N\}$, is m^i-strongly convex, differentiable, and its gradient is M^i-Lipschitz on $\mathbb{R}^d$. For $m_T = \min\{m^1, \ldots, m^N\}$ and $M_T = \max\{M^1, \ldots, M^N\}$, let $\beta > 0$ be such that

$$\alpha^2(\phi+1)m_T + 9\alpha\beta\hat{\lambda}_2\phi - 4\alpha^2 M_T^2 - 4\alpha^2(\phi+1)^2 > 0, \tag{3}$$

is satisfied for some $\phi > 0$ with $\phi + 1 > \frac{4M_T^2}{m_T}$. Then, for any $\alpha > 0$ and each $i \in \{1, \ldots, N\}$, the algorithm (1) over $\mathcal{G}$ makes $\mathbf{x}^i(t) \to \mathbf{x}^\star$ exponentially fast as $t \to \infty$, starting from initial conditions $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ with $\sum_{i=1}^{N} \mathbf{v}^i(0) = \mathbf{0}_d$.

In Theorem 4.1, note that the requirement $\sum_{i=1}^{N} \mathbf{v}^{\mathbf{i}}(0) = \mathbf{0}_d$ is trivially satisfied by each agent with the choice $\mathbf{v}^i(0) = \mathbf{0}_d$. This is an advantage with respect to the continuous-time coordination algorithms proposed in [12], which requires the nontrivial initialization $\sum_{i=1}^{N} \nabla f^i(\mathbf{x}^i(0)) = \mathbf{0}_d$, and in [11], which requires the initialization on a state communicated among neighbors and is thus subject to communication error.

Remark 4.1 (Role of the design parameters in (1)): We provide here several observations regarding the role of the design parameters α and β. First, note that there always exists β satisfying (3) (for example, any $\beta > 4(\phi+1)^2\alpha/(9\phi\hat{\lambda}_2)$). We have observed in simulation that (3) is only a sufficient condition for many cases, e.g., in the numerical example reported here the algorithm (1) converges for any positive α and β. We can interpret $\alpha > 1$ and $\beta > 1$ as a way of increasing the strong convexity coefficient of the local cost functions and the graph connectivity, respectively. Thus, we can expect that the rate of convergence of the algorithm is increase with higher values of α, β. Our simulations have confirmed this conjecture. The relationship between these parameters and the rate of convergence of the algorithm (1) is more evident in the case of quadratic local cost functions $f^i(\mathbf{x}) = \frac{1}{2}(\mathbf{x}^\top\mathbf{x} + \mathbf{x}^\top\mathbf{a}^i + \mathbf{b}^i)$, $i \in \{1, \ldots, N\}$. In this case, the algorithm (1) is a linear time-invariant system where the eigenvalues of the system matrix are $-\alpha$, with multiplicity of Nd, and λ_i, $i \in \{1, \ldots, N\}$ (λ_i's are the eigenvalues of $\mathbf{L}$), with multiplicity d. Therefore, one can show that (1) converges regardless of the value of α, $\beta > 0$ with an exponential rate equal to $\min\{\alpha, \beta\Re(\lambda_2)\}$. •

Next, we study the convergence of (1) over dynamically changing, strongly connected, and weight-balanced digraphs with uniformly bounded and piecewise constant adjacency matrices. The proof of Theorem 4.1 relies on a Lyapunov function with no dependency on the system parameters and its derivative is upper bounded by a quadratic negative definite function. As such, we can readily extend the convergence result to dynamically changing networks.

Proposition 4.1 (Convergence of (1) over dynamically changing interaction topologies): Let $\mathcal{G}$ be a time-varying digraph which is strongly connected and weight-balanced at all times and whose adjacency matrix is uniformly bounded and piecewise constant. Assume the local cost function f^i, $i \in \{1, \ldots, N\}$, is m^i-strongly convex, differentiable, and its gradient is M^i-Lipschitz on $\mathbb{R}^d$. Let $\beta > 0$ satisfy (3) with $\hat{\lambda}_2$ replaced by $(\hat{\lambda}_2)_{\min} = \min_{p \in \mathcal{P}}\{\hat{\lambda}_2(\mathbf{L}_p)\}$, where $\mathcal{P}$ is the index set of all possible realizations of $\mathcal{G}$. Then, for any $\alpha > 0$ and each $i \in \{1, \ldots, N\}$, the algorithm (1) over $\mathcal{G}$ makes $\mathbf{x}^i(t) \to \mathbf{x}^\star$ exponentially fast as $t \to \infty$, starting from initial conditions $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ with $\sum_{i=1}^{N} \mathbf{v}^i(0) = \mathbf{0}_d$.

We conclude this section by analyzing the privacy preservation properties of the algorithm (1). More specifically, we characterize the topological requirements on the communication graph and the knowledge about the algorithm's parameters and initial conditions that allow an agent to reconstruct the local gradients of other agents in the network.

Proposition 4.2 (Privacy preservation under (1)): Let $\mathcal{G}$ be a strongly connected and weight balanced digraph. For $\alpha, \beta > 0$, consider any execution of the coordination algorithm (1) over $\mathcal{G}$ starting from $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ with $\sum_{i=1}^{N} \mathbf{v}^i(0) = \mathbf{0}_d$. Then, an agent $i \in \{1, \ldots, N\}$ can reconstruct the local gradient of another agent $j \neq i$ only if j and all its out-neighbors are out-neighbors of i, and agent i knows $\mathbf{v}^j(0)$ and a_{jk}, $k \in \{1, \ldots, N\}$ (here we assume that the agent i is aware of the identity of neighbors of agent j

and it has memory to save the time history of the data it receives from its out-neighbors).

The requirements of Proposition 4.2 are trivially satisfied when agent i is aware that it is the only out-neighbor of j and all agents know that the algorithm is initialized with $\mathbf{v}^j(0) = \mathbf{0}_d$, for all $j \in \{1, \dots, N\}$.

B. Connected Undirected Graphs

Here, we study the convergence of the algorithm (1) over connected undirected graph topologies. While the results of the previous section are of course valid for these topologies, here using the structural properties of the Laplacian matrix we establish the convergence of (1) for a larger family of local cost functions. We are also able to analytically establish convergence for any $\alpha, \beta > 0$, as we show next.

Theorem 4.2 (Exponential convergence of (1) over connected graphs): Let $\mathcal{G}$ be a connected graph. Assume the local cost function f^i, $i \in \{1, \dots, N\}$, is m^i-strongly convex and differentiable on $\mathbb{R}^d$, and its gradient is locally Lipschitz. Then, for any $\alpha, \beta > 0$ and each $i \in \{1, \dots, N\}$, the algorithm (1) over $\mathcal{G}$ satisfies $\mathbf{x}^i(t) \to \mathbf{x}^\star$ exponentially fast as $t \to \infty$, starting from initial conditions $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ with $\sum_{j=1}^N \mathbf{v}^i(0) = \mathbf{0}$.

Note that the requirement that ∇f^i is locally Lipschitz is trivially satisfied if f^i is twice differentiable. Next, we study the convergence of (1) over connected graphs when the local cost functions are only convex. Here, the lack of strong convexity makes us rely on a LaSalle function to establish asymptotic convergence to the optimizer.

Theorem 4.3 (Asymptotic convergence of (1) over connected graphs): Let $\mathcal{G}$ be a connected graph. Assume the local cost function f^i, $i \in \{1, \dots, N\}$, is convex and differentiable on $\mathbb{R}^d$, and the global cost function f is strictly convex and differentiable on $\mathbb{R}^d$. Then, for any $\alpha, \beta > 0$ and each $i \in \{1, \dots, N\}$, the algorithm (1) over $\mathcal{G}$ satisfies $\mathbf{x}^i(t) \to \mathbf{x}^\star$ as $t \to \infty$, starting from any initial conditions $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ with $\sum_{i=1}^N \mathbf{v}^i(0) = \mathbf{0}_d$.

Remark 4.2 (Simplification of (1) for strictly convex local cost functions): Using the LaSalle function identified in the proof of Theorem 4.3, one can show that the algorithm

$$\dot{\mathbf{v}}^i = \sum_{j=1}^N \mathsf{a}_{ij}(\mathbf{x}^i - \mathbf{x}^j),$$
$$\dot{\mathbf{x}}^i = -\nabla f^i(\mathbf{x}^i) - \mathbf{v}^i,$$

over a connected graph is also guaranteed to asymptotically converge to the optimizer starting from any initial conditions $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ with $\sum_{i=1}^N \mathbf{v}^i(0) = \mathbf{0}_d$ if the local cost functions are strictly convex. •

V. Continuous-time Evolution with Discrete-Time Communication

The implementation of (1) requires continuous-time communication among the agents. While this abstraction is useful for analysis, in practical scenarios the communication service is only available at discrete instants of time. This observation motivates our study here of discrete-time communication implementations of the algorithm (1). Throughout the section, we deal with communication topologies described by connected undirected graphs. In our developments below, we assume synchronous communication across the network. We start by introducing some useful conventions. At any given time $t \in \mathbb{R}_{\geq 0}$, let $\hat{\mathbf{x}}^j$ be the last known state of agent $j \in \{1, \dots, N\}$ transmitted to its in-neighbors. If $\{t_k\} \subset \mathbb{R}_{\geq 0}$ denotes the times at which agents communicates with their in-neighbors, then one has $\hat{\mathbf{x}}^i = \mathbf{x}^i(t_k)$ for $t \in [t_k, t_{k+1})$. Consider the next implementation of the algorithm (1) with discrete-time communication,

$$\dot{\mathbf{v}}^i = \alpha\beta \sum_{j=1}^N \mathsf{a}_{ij}(\hat{\mathbf{x}}^i - \hat{\mathbf{x}}^j), \tag{5a}$$

$$\dot{\mathbf{x}}^i = -\alpha \nabla f^i(\mathbf{x}^i) - \beta \sum_{j=1}^N \mathsf{a}_{ij}(\hat{\mathbf{x}}^i - \hat{\mathbf{x}}^j) - \mathbf{v}^i. \tag{5b}$$

Clearly, the evolution of (5) depends on the sequences of communication times for the agents. Here, we consider two scenarios. Section V-A studies periodic communication schemes where all agents communicate synchronously at fixed Δ intervals of time, i.e., $t_k = \Delta k$. We provide a characterization of the periods that guarantee the asymptotic convergence of (5) to the optimizer. In general, periodic schemes might result in a wasteful use of the communication resources because of the need to account for worst-case situations in determining appropriate periods. This motivates our study in Section V-B of event-triggered communication schemes that tie the communication times to the network state for greater efficiency. Here, we discuss a synchronous centralized event-triggered communication implementation and refer the reader to [15] for a distributed asynchronous implementation. We pay special attention to ruling out the presence of Zeno behavior (the existence of an infinite number of updates in a finite interval of time).

A. Periodic Communication

The following result provides an upper bound on the size of admissible stepsizes for the execution of (5) over connected graphs with periodic communication schemes.

Theorem 5.1 (Convergence of (5) with periodic communication): Let $\mathcal{G}$ be a connected graph. Assume the local cost function f^i, $i \in \{1, \dots, N\}$, is m^i-strongly convex, differentiable, and its gradient is M^i-Lipschitz on $\mathbb{R}^d$. Given $\alpha, \beta > 0$, consider an implementation of the algorithm (5) with agents communicating over $\mathcal{G}$ synchronously every Δ seconds starting at $t_1 = 0$, i.e., $t_k = \Delta k$ for all $i \in$

$\{1, \dots, N\}$. *Let* $0 < \epsilon < 1$ *and* $\delta > 0$ *such that*

$$\phi + 1 = \frac{1}{2m_T} M_T^2 + \frac{1}{2m_T \alpha^2} \delta > 1, \quad (6)$$

where M_T *and* m_T *are given in the statement of Theorem 4.1, and define*

$$\tau = \frac{1}{\alpha M_T + 1} \ln\Big(1 + \frac{(\alpha M_T + 1)\zeta}{\alpha M_T + 1 + \beta \lambda_N \sqrt{1 + \alpha^2}(1 + \zeta)}\Big), \quad (7)$$

where $\zeta^2 = \frac{2\epsilon\lambda_2 \min\{1-\epsilon, \delta\}}{\alpha\beta\lambda_N^2 \phi + 2\alpha^2\lambda_2(1+\phi)^2}$. *Then, if* $\Delta \in (0, \tau)$, *the algorithm evolution starting from initial conditions* $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ *with* $\sum_{i=1}^N \mathbf{v}^i(0) = \mathbf{0}_d$ *makes* $\mathbf{x}^i(t) \to \mathbf{x}^\star$ *exponentially fast as* $t \to \infty$, *for all* $i \in \{1, \dots, N\}$.

Remark 5.1 (Dependence of the communication period on the design parameters): It is interesting to note that the value of τ in Theorem 5.1 depends on the graph topology, the parameters of the local cost functions, the algorithm design parameters α and β, and the variables ϵ and δ. One can use this dependency to maximize the value of τ. Notice that the argument of $\ln(.)$ in (7) is a monotonically increasing function of $\zeta > 0$. Therefore, the smaller the value of β, the larger the value of τ. However, the dependency of τ on the rest of the parameters listed above is more complex. For given local cost functions, fixed network topology and fixed values of α, β, the maximum value of ζ is when $\phi + 1$ is at its minimum and $\epsilon\lambda_2 \min\{1 - \epsilon, \delta\}$ is at its maximum. •

B. Centralized Event-Triggered Communication

This section studies the design of a synchronous centralized event-triggered communication scheme for the algorithm (5). In contrast to periodic schemes, event-triggered implementations tie the determination of the communication times to the current network state, resulting in a more efficient use of the resources. Our discussion builds upon the examination of the Lie derivative of the Lyapunov function used in the proof of Theorem 5.1. In fact, the Lie derivative along (5) is negative definite for all $t \geq 0$ if we ensure that

$$\|\mathbf{\Pi}(\mathbf{x}(t_k) - \mathbf{x})\| \leq \zeta \sqrt{\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \|\mathbf{\Pi}(\mathbf{v} - \bar{\mathbf{v}})\|^2}. \quad (8)$$

Then τ, given in (7), is a lower bound on the time it takes for $\|\mathbf{\Pi}(\mathbf{x}(t_k) - \mathbf{x})\| / \sqrt{\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \|\mathbf{\Pi}(\mathbf{v} - \bar{\mathbf{v}})\|^2}$ to evolve from zero to ζ. The reason that we cannot employ (8) directly as an event-triggered communication law is the lack of knowledge of the solution $\mathbf{x}^\star$ of the optimization problem. We can show that the Lie derivative of the Lyapunov function identified in the proof of Theorem 5.1 is negative definite also when

$$\|\mathbf{\Pi}(\mathbf{x}(t_k) - \mathbf{x}(t))\|^2 \leq \kappa \|\mathbf{\Pi}\mathbf{x}(t)\|^2, \quad t \geq 0, \quad (9)$$

where κ is shorthand notation for

$$\kappa = 2\frac{\epsilon\delta\lambda_2 + 2\phi\alpha\beta\lambda_2^2\epsilon^2(1-\epsilon)}{\alpha\beta\phi\lambda_N^2 + 2\lambda_2\alpha^2(1+\phi)^2}, \quad (10)$$

(here $0 < \epsilon < 1$ and ϕ is given by (6)). Notice that this condition can be evaluated without the knowledge of the solution $\mathbf{x}^\star$ of the optimization problem. According to the above discussion, the sequence of synchronous communication times $\{t_k\}_{k\in\mathbb{N}} \subset \mathbb{R}_{\geq 0}$ for (5) should be determined by (9). However, for a truly implementable law, one should guarantee that no Zeno behavior occurs, i.e., the sequence of times does not have any finite accumulation point. However, observing (9), one can see that Zeno behavior will arise at least near the agreement surface $\mathbf{\Pi}\mathbf{x} = \mathbf{0}_{dN}$. The following result details how we address this problem to design a Zeno-free centralized event-triggered communication law.

Theorem 5.2 (Convergence of (5) with Zeno-free centralized event-triggered communication): Let $\mathcal{G}$ *be a connected graph. Assume the local cost function* f^i, $i \in \{1, \dots, N\}$, *is* m^i*-strongly convex, differentiable, and its gradient is* M^i*-Lipschitz on* $\mathbb{R}^d$. *Consider an implementation of the algorithm (5) with agents communicating over* $\mathcal{G}$ *synchronously at times* $\{t_k\}_{k\in\mathbb{N}} \subset \mathbb{R}_{\geq 0}$, *starting at* $t_1 = 0$, *determined by*

$$t_{k+1} = \operatorname{argmax}\{t \in [t_k + \tau, \infty) \mid \|\mathbf{\Pi}(\mathbf{x}(t_k) - \mathbf{x}(t))\|^2 \leq \kappa \|\mathbf{\Pi}\mathbf{x}(t)\|^2\}, \quad (11)$$

where τ *and* $\kappa < 1$ *are defined in (7) and (10), respectively. Then, for any given* $\alpha, \beta > 0$ *and each* $i \in \{1, \dots, N\}$, *the algorithm evolution starting from initial conditions* $\mathbf{x}^i(0), \mathbf{v}^i(0) \in \mathbb{R}^d$ *with* $\sum_{i=1}^N \mathbf{v}^i(0) = \mathbf{0}_d$ *makes* $\mathbf{x}^i(t) \to \mathbf{x}^\star$ *exponentially fast as* $t \to \infty$.

Interestingly, given that (9) does not use the full state of the network but rather relies on the computation of disagreement, one can interpret it as an output feedback event-triggered controller. Guaranteeing the existence of lower bounded inter-execution times for such controllers is in general a difficult problem, see e.g., [16]. Augmenting (9) with the condition $t_{k+1} \geq t_k + \tau$ results in Zeno-free executions by lower bounding the inter-event times by τ. The knowledge of this value also allows a designer to compute bounds on the maximum energy spent by the network on communication during any given time interval.

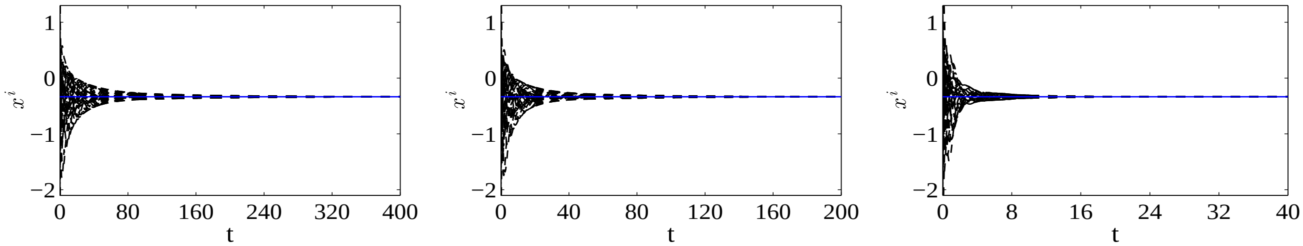
VI. Simulations

Consider a network of 30 agents, where the local cost function of agent i is given by

$$f^i(x) = 0.5(x + e^i)^2 + c^i \mathrm{e}^{-a^i x} + d^i \mathrm{e}^{-b^i x}, \; i \in \{1, \dots, N\}.$$

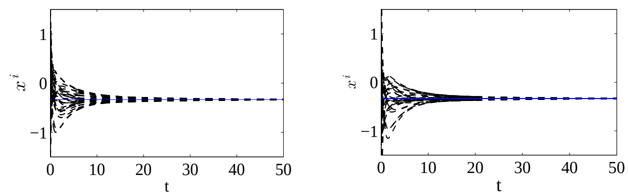
The coefficients are chosen randomly uniformly as a^i, b^i, c^i, $d^i \sim \mathcal{U}[0.1, 1]$ and $e^i \sim \mathcal{U}[-1, 2]$, Figure 1 illustrates the performance of the algorithm (1) over a ring digraph whose edges change direction with time multiple times. Convergence is achieved as guaranteed by Proposition 4.1. The plot also shows that larger values of β result in faster convergence, cf. Remark 4.1. In all our simulations of this example, convergence is achieved for any $\alpha, \beta > 0$.

Figure 2(a)-(b) compares the performance the algorithm (5) employing a periodic communication with the performance of the continuous-time algorithm (1) over an undirected connected ring communication graph. It is interesting to note the comparable performance between both algorithms observed in these plots.



(a) $\alpha = 1$, $\beta = 0.5$ (b) $\alpha = 1$, $\beta = 1$ (c) $\alpha = 1$, $\beta = 5$

Fig. 1: Executions of the algorithm (1) over a ring digraph whose direction changes every 5 seconds (weights are unitary): the dashed lines (resp. solid blue line) show the time history of x^i's (resp. $\mathsf{x}^\star$).



(a) Algorithm (5) with $\alpha = 1$, $\beta = 2$, $\Delta = 0.15$ sec. (b) Algorithm (1) with $\alpha = 1$, $\beta = 2$

Fig. 2: Performance comparison between the algorithms (5) and (1) over an undirected ring communication graph: the dashed lines (resp. solid blue line) show the time history of x^i's (resp. $\mathsf{x}^\star$).

VII. Conclusions

We have proposed a distributed continuous-time optimization algorithm for a network with a strongly connected and weight-balanced interaction topology and a strictly convex global cost function which is the sum of local cost functions. When the local cost functions are m-strongly convex and their gradients are globally Lipschitz, we have established that the algorithm converges exponentially fast. This property is preserved in dynamic networks as long as the topology stays strongly connected and weight-balanced. For connected undirected graphs, we have proved that the exponential convergence also holds when the local cost functions are m-strongly convex and their gradients are only locally Lipschitz. For such networks, we also showed that when the local cost functions are convex the proposed algorithm converges asymptotically. We have also investigated the discrete-time implementation of our algorithm, providing an upper bound on the suitable stepsize for connected graphs, and designing a centralized, Zeno-free event-triggered implementation. Finally, we have characterized the privacy preservation properties of our algorithm. Future work will focus on pursuing the design of distributed event-triggered implementations and the use of triggered control methods in other distributed optimization and coordination problems, including constrained, time-varying, and online scenarios, and networked games.

Acknowledgments

This work was supported by L3 Communications through the UCSD Cymer Center for Control Systems and Dynamics and NSF Award ECCS-1307176.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[2] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Transactions on Automatic Control*, vol. 31, no. 9, pp. 803–812, 1986.

[3] N. A. Lynch, *Distributed Algorithms*. Morgan Kaufmann, 1997.

[4] M. G. Rabbat and R. D. Nowak, "Distributed optimization in sensor networks," in *Symposium on Information Processing of Sensor Networks*, pp. 20–27, April 2004.

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

[6] P. Wan and M. Lemmon, "Event-triggered distributed optimization in sensor networks," in *Symposium on Information Processing of Sensor Networks*, (San Francisco, CA), pp. 49–60, 2009.

[7] A. Nedić, A. Ozdaglar, and P. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 922–938, 2010.

[8] M. Zhu and S. Martínez, "On distributed constrained formation control in operator-vehicle adversarial networks," *Automatica*, 2013. In press.

[9] J. Wang and N. Elia, "A control perspective for centralized and distributed convex optimization," in *IEEE Conf. on Decision and Control*, (Florida, USA), December 2011.

[10] B. Gharesifard and J. Cortés, "Distributed continuous-time convex optimization on weight-balanced digraphs," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 781–786, 2014.

[11] F. Zanella, D. Varagnolo, A. Cenedese, G. Pillonetto, and L. Schenato, "Newton-Raphson consensus for distributed convex optimization," in *IEEE Conf. on Decision and Control*, (Florida, USA), pp. 5917–5922, December 2011.

[12] J. Lu and C. Y. Tang, "Zero-gradient-sum algorithms for distributed convex optimization: The continuous-time case," *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2348–2354, 2012.

[13] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: regrets and intrinsic privacy-preserving properties," 2012. To appear (available online).

[14] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*. American Mathematical Society, Princeton University Press, 2009. Available at http://www.coordinationbook.info.

[15] S. S. Kia, J. Cortés, and S. Martínez, "Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication," *Automatica*, 2014. Submitted.

[16] M. Donkers and W. Heemels, "Output-based event-triggered control with guaranteed $\mathcal{L}_\infty$-gain and improved and decentralized event-triggering," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1362–1376, 2012.