

Convex Decreasing Algorithms: Distributed Synthesis and Finite-time Termination in Higher Dimension

James Melbourne, Govind Saraswat, Vivek Khatana, Sourav Patel, and Murti V. Salapaka *Fellow, IEEE*

Abstract—We establish finite time termination algorithms for consensus algorithms based on geometric properties that yield finite-time guarantees, suited for use in high dimension and in the absence of a central authority. These pursuits motivate a new peer to peer convex hull algorithm which is utilized for one stopping algorithm. Further an alternative lightweight norm based stopping criteria is also developed. The practical utility of the algorithm is illustrated through MATLAB simulations.

Index Terms—Distributed Consensus, Multi-agent systems, Network-based computing systems, High-dimensional state algorithms, Convex hull

I. INTRODUCTION

Consensus algorithms are a fundamental tool utilized in multiple disciplines, in particular they are used for maintaining the reliability of systems in distributed and multi-agent contexts. The ideas of distributed consensus algorithms can be traced back to the seminal works, see [2]–[5]. We direct the reader to [6], [7] for background on the topic in the context of control. For additional background, recent work and motivation for treating the high-dimensional case, see [8]–[18].

In practice, all consensus algorithms must terminate in finite time, and more rapid termination saves valuable computation and communication resources. The purpose of this paper is to provide distributed stopping mechanisms for an $\mathbb{R}^n$ -valued consensus algorithm in a distributed manner when all agents are within an ε -threshold of consensus.

To do so we introduce and analyze two distributed algorithms for diameter estimation, that provide upper bounds on a

system of agents. When the agents behavior, consensus or not, satisfies a geometric property we call “convex decreasing”, it is shown diameter bounds that hold at t_0 holds for all time $t \geq t_0$. In particular for a consensus algorithm, every agents is, and will remain for all time within ε of consensus. Our notion of an ε norm is flexible with respect to choice of norm. When agents take values in $\mathbb{R}^n$, there is thus considerable diversity in the meaning of an ε -threshold, as there is considerable diversity of norms, even among ℓ_p . For instance a vector x satisfying a ε -threshold for ℓ_∞ in the sense that $\|x\|_\infty := \max_i |x_i| < \varepsilon$ would only guarantee that $\|x\|_2 := \sqrt{\sum_i |x_i|^2} < \sqrt{n}\varepsilon$. Thus in high dimension, using the ℓ_∞ norm to determine the ℓ_2 convergence (or vice versa) comes with high cost. As such we emphasize that our approach is flexible enough to be implemented for convergence with respect to an arbitrary norm.

A precise definition of a convex decreasing algorithm will be given in Section II. For now we remark that such algorithms ensure a non-increasingness of the diameter of the values of the agents and are ubiquitous in consensus. Convex decreasing algorithms can be understood as a relaxation of the class of algorithms studied by Moreau in [19], and hence the stopping algorithms introduced here may be immediately applied. Additionally, the non-linear “push-sum” (as referred to in [8]) or “ratio-consensus” algorithm (as it is referred to in [20]) is proven to be convex decreasing here, even in the case of time-dependent, and randomly chosen update matrices, extending the work of the authors in [21] which treated the deterministic, time homogeneous case. This gives a rigorous mathematical explanation for the relative robustness and stability of ratio consensus enjoyed even in complicated practical implementations. We thank an anonymous reviewer for suggesting [22] where the “circumcenter algorithm” provides another example of a convex decreasing algorithm.

As mentioned we introduce two algorithms for distributed diameter estimation. The first is based on the determination of extreme points of the values taken by the agents. In short agents share their best knowledge of the extreme values of the convex hull of the groups values. The agents are able to reconstruct the extreme values within D -iterations. Determining the extreme points of a set in a distributed fashion is tantamount to a distributed convex hull algorithm and the is of independent interest beyond diameter estimation. With knowledge of the extreme points in hand an agent can reassemble the convex hull and make many useful

This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the Advanced Research Projects Agency-Energy (ARPA-E) under grant no. DE-AR0000701. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes. A draft of the paper has been made available online [1].

James Melbourne (email: james.melbourne@ciat.mx), Vivek Khatana (khata010@umn.edu), Sourav Patel (patel292@umn.edu), and Murti V. Salapaka (murtis@umn.edu) are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, USA.

Govind Saraswat is with the National Renewable Energy Laboratory, Golden, CO, USA (govind.saraswat@nrel.gov).

computations. Most relevant to our efforts here, the agents can upper bound the value of a convex function on the convex hull of the extreme points. In particular the distances of the algorithm from consensus with respect to any norm can be controlled from above.

When agents take scalar values, this reduces to maxima and minima protocols already utilized in the consensus literature. See [23] where this property was leveraged toward finite time stopping for consensus when agents update to convex combinations of their neighbors values. The respective decreasingness and increasingness of maxima and minima protocols mathematically legitimizes their use for stopping, and corresponds to be introduced notion of convex decreasingness of the consensus algorithm in the one dimensional case. This monotonicity was proven for the maxima and minima for the one dimensional ratio consensus algorithm in [24] and utilized for finite stopping. This was then extended to time dependent cases in [25], [26]. By treating the coordinates of an n -dimensional ratio consensus algorithms as one dimensional ratio consensus algorithms an n -dimensional stopping algorithm was introduced in [24]. The work here contains all the aforementioned as special cases and in particular provides a considerable sharpening both theoretically and in application, of [24].

The second method for stopping is based on agents passing a auxiliary "radius" value. This stopping algorithm is flexible to any choice of norm, however unlike the first, the norm must be chosen before initialization of the algorithm. Though both algorithms, are based on distributed computations, and scale with the level of connectivity at a node (agnostic to the network), the radius based stopping algorithm has the advantage of having a computational cost at a node that grows linearly in the number of neighbors. Additionally it has a communication cost of a single scalar value, independent of the size of the network and the dimension of the consensus values, making it dramatically more efficient than other proposed algorithms at high dimension.

We mention that for small networks, the authors in [27] present a method terminating in a finite number of iterations. However, [27] requires each node to run n (total number of agents) different linear iterations each for at least $n + 1$ time-steps. The algorithms for stopping developed here have computational cost on the order of neighbors to a node. Thus the algorithms presented here provide significant improvement, for example, in large networks with sparse connectivity.

Let us note that the diameter estimation algorithms terminate within D -iterations and provide valid bounds independent of convergence of said agents values. This is especially significant in real world applications, where theoretical guarantees can be intractable, probabilistic, or practically inefficient. The algorithms here have the advantage of addressing the particular realization of the system, and not a theoretical worst case.

Statement of contribution:

- 1) This article introduces two new algorithms for distributed stopping of $\mathbb{R}^n$ -valued consensus algorithms within an ϵ -threshold of consensus with respect to arbitrary norm.
- 2) The algorithms correspond to distributed diameter es-

timiation methods that are of independent interest. In particular the first diameter estimation algorithm is based on distributed convex hull determination, which is of broad utility. The second introduced algorithm is based on a radius sharing technique and has minimal communication cost of the stopping algorithm, requiring only a single scalar value to be shared at each iteration. This is especially significant as previous stopping algorithms carried a communication cost of order n scalars, where n equal to the dimension.

- 3) We introduce a geometric notion, termed "convex decreasing" which ensures that the diameter bounds obtained, yield guarantees on the distance to consensus. Moreover we demonstrate that this property holds for many widely utilized consensus algorithms. In particular, the convex monotonicity proven for the deterministic and time homogeneous, ratio consensus algorithm in [21] is extended to treat the case of random and time dependent updates.

The techniques here can also be used to obtain new stopping criteria for consensus based least square estimation, as well as distributed function calculation. For brevity of exposition, such applications of the theory developed will be reported elsewhere.

The paper is organized as follows. In Section II, the basic definitions needed for subsequent developments are presented. Further, we discuss the setup for the distributed average consensus in higher dimensions (called the vector consensus problem) using ratio consensus. Section IV presents an analysis on the polytopes of the network states generated in the ratio consensus algorithm. Section IV-B establishes a norm-based finite-time termination criterion for the vector consensus problem. Theoretical findings are validated with simulations presented in Section V followed by conclusions in Section VI.

II. PRELIMINARIES

In this section we present definitions and elementary results essential for the subsequent developments. Detailed description of graph theory and linear algebra notions are available in [28], and [29] respectively.

Definition 1: (Directed Graph) A directed graph (denoted as digraph) $\mathcal{G}$ is a pair $(\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is a set of vertices or nodes and $\mathcal{E}$ is a set of edges, (i, j) which are ordered subsets elements of $\mathcal{V}$.

We use standard graph theoretic terminology, recalled here. In a directed graph, a directed path from node i to j exists if there is a sequence of distinct directed edges of $\mathcal{G}$ of the form $(k_1, i), (k_2, k_1), \dots, (j, k_m)$. For the rest of the article, a path refers to a directed path. The path length, or length of a path is the number of directed edges belonging to the path. By convention, we consider a node $i \in \mathcal{V}$ to be connected to itself by a path of length zero. The diameter¹ of a graph is the longest shortest path between any two nodes in the network. Unless stated otherwise we assume there exists an upper bound

¹When a graph is undirected, path length defines a metric, and this coincides with the usual notion of diameter. That is, for S a set with metric d , the $\text{diam}(S) = \sup_{x, y \in S} d(x, y)$.

D on the diameter of the graph throughout the rest of the article.

We note that the notion of diameter can be extended to the case that a network takes finitely many $\{\mathcal{E}_1, \dots, \mathcal{E}_m\}$ different topologies determined by a random process $\sigma(k)$ taking values in $\{1, \dots, m\}$ such that the topology of the network at time t is given by $\mathcal{E}_{\sigma(t)}$. In this case we consider the diameter of the network D to be the minimum over $\hat{D}$ where $\hat{D}$ is an integer such that given $i, j \in \mathcal{V}$ and a sequence $n_1, \dots, n_{\hat{D}}$ such that $\mathbb{P}((\sigma(1), \dots, \sigma(\hat{D})) = (n_1, \dots, n_{\hat{D}})) > 0$, then there exists a sequence $k_0, k_1, \dots, k_m, k_{m+1}$ in $\mathcal{V}$ with $m < \hat{D}$, $i = k_0$ and $j = k_{m+1}$ and $(k_l, k_{l-1}) \in \mathcal{E}_{\sigma(l)}$. In short, the diameter is the longest duration of time it can take to travel from one node to another. We make no assumptions on the distribution of the process σ , except that $D < \infty$. For simplicity of exposition, we will not further emphasize the technicalities of random topology switching. We will denote the cardinality of a set A by $|A|$.

Definition 2: (m -In-Neighborhood) For $m = 0$ define $N_i^-(0) = i$, and for $m > 0$ define $N_i^-(m)$ to be $\bigcup_{j \in N_i^-(m-1)} N_j^-$, so that $N_i^-(m)$ is the set of nodes $j \in G$ from which i can be reached in m or less steps. When $m = 1$ we write N_i^- , called the In-neighborhood of i .

Definition 3: (Network and Consensus algorithms) A network algorithm on $G = G(\mathcal{V}, \mathcal{E})$, is a finite or countably infinite sequence of maps $\phi : W^{\mathcal{V}} \rightarrow W^{\mathcal{V}}$ called network updates for a normed space $(W, \|\cdot\|)$, a consensus algorithm is a network algorithm such that for any state $f \in W^{\mathcal{V}}$, $\Phi_n(f) := \phi_n \circ \phi_{n-1} \circ \dots \circ \phi_1(f)$ satisfies, $\lim_{n \rightarrow \infty} \|\Phi_n(f)(i) - \Phi_n(f)(j)\| = 0$, for all $i, j \in \mathcal{V}$, where ϕ_i are network updates.

We will have particular interest in distributed consensus algorithms that can be build from matrix operations. We recall the following definitions

Definition 4: (Stochastic Matrices) A real $N \times N$ matrix $\mathbf{P} = [p_{ji}]$ is called a column stochastic matrix if $p_{ji} \geq 0$ for $1 \leq i, j \leq N$ and $\sum_{j=1}^N p_{ji} = 1$ for $1 \leq i \leq N$. A real $N \times N$ matrix $\mathbf{A} = [a_{ij}]$ is called a row stochastic matrix if $1 \geq a_{ij} \geq 0$ for $1 \leq i, j \leq N$ and $\sum_{j=1}^N a_{ij} = 1$ for $1 \leq i \leq N$.

As is well known, both row stochastic or column stochastic updating matrices can be constructed by agents locally. However, constructing an updating matrix that is simultaneously row and column stochastic (known as doubly stochastic), though desirable for its simple convergence properties, is not possible without global topological knowledge.

As a notational convention, matrices will be written in bold face as above. One can identify the vector space W with $\mathbb{R}^d$ without any loss of generality.

Definition 5: (Convex hull) For a set $U \subseteq W$, the convex hull of U is the smallest convex set containing U ,

$$\text{co}(U) = \bigcap_{\{F \text{ convex } : U \subseteq F\}} F. \quad (1)$$

The closure of a set U , denoted $\bar{U}$, is the smallest closed set containing U . We write the closure of a convex hull by $\bar{\text{co}}(U) := \text{co}(\bar{U})$.

Compact, convex sets are characterized by their extreme points, whose definition we recall now.

Definition 6: (Extreme point) For a convex set $U \subseteq W$ define $u \in U$ to be an extreme point of U , denoted $\mathcal{E}(U)$, if $u = \frac{u_1 + u_2}{2}$ for $u_i \in U$ implies $u_1 = u_2 = u$. For a general U , define $\mathcal{E}(U) := \mathcal{E}(\text{co}(U))$.

Definition 7: (Convex Decreasing) A sequence of sets $S_n \subseteq W$ is convex decreasing if $\text{co}(S_{n+1}) \subseteq \text{co}(S_n)$. A W consensus algorithm is convex decreasing when the sets $S_n = \Phi_n(f)$ are convex decreasing for any $f \in W^{\mathcal{V}}$.

Definition 8: (Support function) For a non-empty set $A \subseteq \mathbb{R}^d$, define its support function

$$h_A(u) := \sup_{x \in A} \langle x, u \rangle,$$

where $\langle x, y \rangle := \sum_{i=1}^n x_i y_i$.

Proposition 2.1: Support functions satisfy the following:

- 1) $A \subseteq B$ implies $h_A \leq h_B$
- 2) $h_A = h_{\text{co}(A)}$
- 3) $h_A = h_{\bar{A}}$
- 4) $h_A \leq h_B$ implies $\bar{\text{co}}(A) \subseteq \bar{\text{co}}(B)$.
- 5) A sequence of compact sets $\{A_n\}$ is convex decreasing if and only if $h_{A_n} \geq h_{A_{n+1}}$ holds for all n .

Proof: (1) - (4) are standard results from convex geometry, see [30].

To prove (5) observe that if $\{A_n\}$ is a convex decreasing sequence, by definition $\text{co}(A_{n+1}) \subseteq \text{co}(A_n)$. So that $h_{A_n} = h_{\text{co}(A_n)} \geq h_{\text{co}(A_{n+1})} = h_{A_{n+1}}$. Conversely, if $h_{A_n} \geq h_{A_{n+1}}$ then by (4), $\bar{\text{co}}(A_{n+1}) \subseteq \bar{\text{co}}(A_n)$, and since the A_n are assumed compact their convex hulls are as well, and hence $\text{co}(A_{n+1}) \subseteq \text{co}(A_n)$. ■

Lemma 2.1: For K convex and compact $\bar{\text{co}}(\mathcal{E}(K)) = K$. For $A_\alpha \subseteq \mathbb{R}^d$, then $\text{co}(\bigcup_\alpha A_\alpha) = \text{co}(\bigcup_\alpha \text{co}(A_\alpha))$

Proof: The first result can be found in [30]. The second is an easy exercise. ■

III. EXAMPLES OF CONVEX DECREASING ALGORITHMS

We give two archetypal examples and note that we need no assumption of connectivity to prove these examples convex decreasing.

Example 1: (Row Stochastic Updating) Let $\mathbf{A}(k)$ be a sequence of (random) $N \times N$ row stochastic matrices. Given an arbitrary initialization $z^i(0)$ of N agents. The sequence $z(k)$ defined by

$$z^i(k+1) = \sum_{j=1}^N A_{ij}(k) z^j(k)$$

is convex decreasing.

Since $z^i(k+1)$ can be written explicitly as $\sum_{j=1}^N A_{ij}(k) z^j(k)$, a convex combination of $\{z^j(k)\}$, by the row stochasticity of $\mathbf{A}(k)$, $z(k+1)$ clearly belongs to the convex hull of $z(k)$. The second main example is more subtle and is included as the following theorem.

Theorem 3.1: (Ratio Consensus) Let $\mathbf{P}(k)$ be a sequence of (random) $N \times N$ column stochastic matrices. Given an

arbitrary initialization of $x^i(0)$ of N agents, initialize an auxiliary sequences, $y^i(0) = 1$, and update by

$$\begin{aligned} x^i(k+1) &= \sum_{j=1}^N P_{ij}(k) x^j(k) \\ y^i(k+1) &= \sum_{j=1}^N P_{ij}(k) y^j(k), \end{aligned}$$

then the sequence

$$r^i(k) = \frac{1}{y^i(k)} x^i(k)$$

is convex decreasing.

Proof: By Proposition 2.1, it is enough to show that the support functions satisfy $h_{r(k+1)} \leq h_{r(k)}$. Since $r^j(k) = x^j(k)/y^j(k)$ the support function satisfies the following inequality for all j ,

$$\langle x^j(k), u \rangle \leq h_{r(k)}(u) y^j(k).$$

With ratio-consensus updates from $\mathbf{P}$ column stochastic, to prove convex decreasingness, since the convex-hull of a compact set K is the intersection of half-spaces containing K , it suffices to show that if $u \in \mathbb{R}^d$ and $c \in \mathbb{R}$ are such that $\langle r^i(k), u \rangle c$ holds for all i , then $\langle r^i(k+1), u \rangle c$ holds for all i as well. Written in terms of support functions, we need to prove, $\langle x^j(k+1), u \rangle \leq h_{r(k)}(u)$, or equivalently,

$$\langle x^j(k+1), u \rangle \leq h_{r(k)}(u) y^j(k+1).$$

Computing,

$$\begin{aligned} \langle x^j(k+1), u \rangle &= \sum_i p_{ji}(k) \langle x^i(k), u \rangle \\ &\leq \sum_i p_{ji} h_{r(k)}(u) y^i(k) \\ &= h_{r(k)}(u) y^j(k+1), \end{aligned}$$

where the equalities follow by definition and by linearity. The inequality used is equivalent to $\langle r^i(k), u \rangle \leq h_{r(k)}(u)$ which follows from the definition of the support function. ■

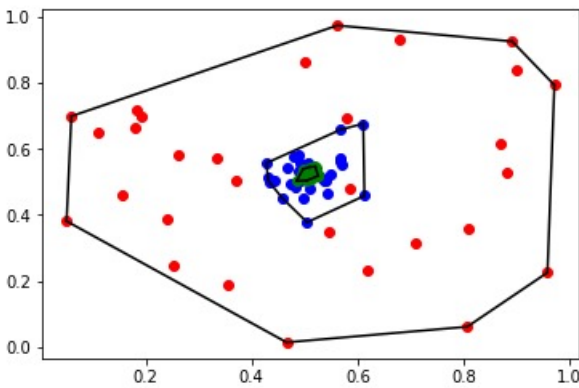


Fig. 1. Iterations of a convex decreasing consensus algorithm. Details on construction in [21].

IV. FINITE-TIME STOPPING CRITERIA

We will use the monotonicity of convex decreasing consensus algorithms to develop a distributed stopping criteria, guaranteeing convergence of all nodes within an ϵ -ball of the consensus value for a general norm.

The consensus value $\bar{x}$ of a convex decreasing consensus algorithm must belong to the convex hull of $\{x_i(t)\}$, see Figure 1. Thus the value of an arbitrary convex function f at $\bar{x}$ can be upper bounded by its maxima on the extreme points of $\{x_i(t)\}$, a set of typically of logarithmic order in the number of agents [31], [32]. In the same spirit, the following demonstrates the monotonicity of diameter with respect to an arbitrary norm.

Theorem 4.1: If $x(t) = \{x_i(t)\}_{i=1}^N$ is convex decreasing in t , and $\|\cdot\|$ is a norm, then $\text{diam}(x(t)) := \sup_{i,j} \|x_i(t) - x_j(t)\|$ is decreasing in t . Moreover, if $x(t)$ is a consensus algorithm, with $\bar{x}$ denoting the consensus value, then $\sup_i \|x_i(t) - \bar{x}\|$ is decreasing in t .

Proof: For any $x \in \mathbb{R}^d$, since $x_i(t+1)$ belongs to the convex hull of $x(t)$, there exists $\lambda_j \geq 0$ such that $\sum_j \lambda_j = 1$ and $x_i(t+1) = \sum_j \lambda_j x_j(t)$, so that $\|x_i(t+1) - x\| = \|\sum_j \lambda_j (x_j(t) - x)\| \leq \sum_j \lambda_j \|x_j(t) - x\| \leq \sup_j \|x_j(t) - x\|$. Repeating the argument gives the first result, taking $x = \bar{x}$ gives the second. ■

Observe that the number of agents used in the diameter computation in Theorem 4.1 can be reduced. One can consider only the set of extreme points of $x(t)$ as

$$\max_{i,j} \|x_i - x_j\| = \max_{x_i, x_j \in \mathcal{E}(x(t))} \|x_i - x_j\|$$

Indeed, for any convex function Ψ and compact set K , $\sup_{x \in K} \Psi(x) = \sup_{x \in \mathcal{E}(K)} \Psi(x)$. In what follows, we develop a distributed convex hull computation algorithm that is of independent interest. For agents $\{1, 2, \dots, N\}$ with states $w(i) \in \mathbb{R}^d$, we wish to communicate the extreme points of w_i and hence the convex hull, across the network in a distributed fashion. We consider hereafter a fixed topology a graph of diameter bounded by D , and an algorithm, where at stage one, agents share their value $w(j)$ with neighbors. The nodes then update their approximation of the convex hull, by determining the extreme points among all values received from their neighbors and their own. This new set of extreme points is communicated to all neighbors and then the process repeats. We show that after D iterations, every node will have determined the extreme points of w .

In the context of a consensus algorithm protocol $w : \mathcal{V} \rightarrow \mathbb{R}^d$ represents $w(i) = r^i(k)$ or $z^i(k)$ the value at node i in iteration k of a consensus algorithm, and we implement the following stopping criterion. Given a norm $\|\cdot\|$ and a tolerance ϵ , implement the convex hull algorithm at time k , then at time $k+D$, at a node j if $\max_{e(i), e(j) \in \mathcal{E}(w)} \|e(i) - e(j)\| \leq \epsilon$, then stop the consensus algorithm. In what follows we demonstrate that upon stopping every node is within ϵ of the consensus value in norm.

A. Peer to Peer Convex Hull Algorithm

We now describe a finite time algorithm for distributed convex hull computation. Suppose that $|\mathcal{V}|$ agents indexed by

i where each agent has a set S_i of elements of $\mathbb{R}^d$. The agent can communicate within the constraints imposed by a specific communication network. We provide a distributed consensus algorithm² through which all agents obtain $\mathcal{E}(\cup_i S_i)$ in D -iterations of the algorithm.

Definition 9: For $S : \mathcal{V} \rightarrow E$, define the initialization $x_i(0) = \mathcal{E}(S_i)$. Iteratively define,

$$x_i(t) = \mathcal{E} \left(\bigcup_{j \in N_i^-(t-1)} x_j(t-1) \right).$$

We identify $x_i(t)$ with an element of E , by writing its elements in lexicographical order. That is, at iteration t , an agent i receives the extreme points known to their “in-neighbors” and forms a new set $s_i(t)$ comprised of their previous extreme points and their neighbors. Agent i finds the extreme points of this new set, and then communicates the set to its “out-neighbors” to initiate another iteration.

Theorem 4.2: For any initial configuration defined by $S : \mathcal{V} \mapsto E$, the algorithm for $x_i(t)$ described in Definition 9 is a distributed consensus algorithm. Moreover, considered as sets

$$x_i(t) = \mathcal{E}(S_i(t)) \quad (2)$$

where $S_i(t) := \cup_{j \in N_i^-(t)} S_j$, and we recall that $N_i^-(t)$ is the t in-neighborhood of node i (see Definition 2). In particular, for $t \geq D$, $x_i(t) = x_i(D) = \mathcal{E}(\cup_{i \in \mathcal{V}} S_i)$.

Proof: The result is true by definition checking when $t = 0$, since $S_i(0) = S_i$. Thus we proceed by induction and assume the result holds for $k < t$. By definition,

$$x_i(t) = \mathcal{E} \left(\bigcup_{j \in N_i^-(t-1)} x_j(t-1) \right). \quad (3)$$

By the induction hypothesis,

$$\begin{aligned} \mathcal{E} \left(\bigcup_{j \in N_i^-(t-1)} x_j(t-1) \right) &= \mathcal{E} \left(\bigcup_{j \in N_i^-(t-1)} \mathcal{E}(S_j(t-1)) \right) \\ &= \mathcal{E} \left(\bigcup_{j \in N_i^-(t-1)} \mathcal{E} \left(\bigcup_{k \in N_j^-(t-2)} S_k \right) \right). \end{aligned}$$

Recall that for non-convex sets U , $\mathcal{E}(U) := \mathcal{E}(\text{co}(U))$, so that

$$\mathcal{E}(\cup_j \mathcal{E}(\cup_{k_j} S_{k_j})) = \mathcal{E}(\text{co}(\cup_j \mathcal{E}(\text{co}(\cup_{k_j} S_{k_j}))))$$

where the subscript k_j ranges over the set $N_j^-(t-1)$. If we write $K_j = \text{co}(\cup_{k_j} S_{k_j})$, and apply Lemma 2.1, with the fact that K is convex and compact,

$$\text{co}(\cup_j \mathcal{E}(K_j)) = \text{co}(\cup_j \text{co}(\mathcal{E}(K_j))) = \text{co}(\cup_j K_j).$$

By definition of K_j and another application of Lemma 2.1,

$$\text{co}(\cup_j K_j) = \text{co}(\cup_j \text{co}(\cup_{k_j} S_{k_j})) = \text{co}(\cup_j \cup_{k_j} S_{k_j}).$$

²The convex hull algorithm can be understood as a distributed consensus algorithm in the space of all finite sequences in $\mathbb{R}^d$.

Thus our result follows once we can show,

$$\bigcup_{j \in N_i^-(1)} \left(\bigcup_{k \in N_j^-(t-1)} S_k \right) = \bigcup_{j \in N_i^-(t)} S_j.$$

Both sets can be considered as unions of S_k indexed by paths of length not larger than t terminating at i . More explicitly, both sets can be written as $\bigcup_{\lambda \in \Lambda} S_\lambda$ where Λ is the space of all paths $v : \{0, 1, 2, \dots, k\} \rightarrow \mathcal{V}$ such that $k \leq t$, $v(k) = v_i$. This gives (2). Since $N_i^-(t) = \mathcal{V}$ for $t \geq D$, $S_i(t) = \cup_{j \in \mathcal{V}} S_j$ and $x_i(D) = x_j(D) = \mathcal{E}(S)$. Thus the algorithm considers is a consensus algorithm. The algorithm is distributed as each $S_i(t)$ is a function of the $S_j(t-1)$ for $j \in N_i^-(1)$. ■

This shows that, agents in a distributed network can obtain exact knowledge of the convex hull in D iterations. As in the discussion in the beginning of this section the convex hull algorithm can be used to provide finite time stopping criterion for a convex decreasing consensus algorithm.

Theorem 4.3: For $z_i(t)$ the state of the i -th agent at time $t \geq 0$ in a convex decreasing consensus algorithm with consensus value $\bar{z}$, then with $S_i = \{z_i(t')\}$, and x_i as in Definition 9,

$$\|\bar{z} - z_i(t)\| \leq \max_{x \in x_j(D)} \|\bar{z} - x\|,$$

for $t \geq t'$.

Proof: By z is convex decreasing, $z_i(t) \in \text{co}(\{z_k(t')\}_k)$. By Theorem 4.2, $\mathcal{E}(\{z_k(t')\}_j) = x_j(D)$ at every j . Since $z \mapsto \|\bar{z} - z\|$ takes its maximum on $\text{co}(\{z_k(t')\}_k)$ at an extreme point of $\{z_k(t')\}$, the theorem follows. ■

Theorem 4.3 demonstrates that an agent can obtain exact bounds on the distance from convergence of the consensus with respect to an arbitrary norm within D iterations of the convex hull algorithm. If $\bar{x}$ denotes the consensus value of the algorithm $x(t)$, obtaining knowledge of the extreme points of $x(t)$ gives decentralized agents upper bounds on $\Phi(\bar{x})$ for any convex function Φ . In particular an agent can compute an upper bound on their own distance from consensus, by choosing $\Phi(x) = \|x - x_i(t)\|$. The significance of this observation, for convex decreasing algorithms, is that an agent will be able to upper bound their own distance from consensus not only for their current state, but for all future states as well.

The distributed convex hull algorithm constructed here is of independent interest. Centralized algorithms for finding the convex hull of a finite set of n points, particularly in $\mathbb{R}^2$, have been long proposed. However, with the recent advancements in distributed multi-agent systems, the problem of estimating the convex hull of set in arbitrary dimension in a distributed manner, has become important, see [33]–[35].

The number of extreme points associated to n points can be as large as n , however, in practice their cardinality is typically much smaller. For example, when the n points are chosen to be iid standard Gaussian, the average number of extreme points is of order, see $\log^{\frac{d-1}{2}}(n)$ [31] and is approximately Gaussian for n large, see [32] with variance no larger than order $\log^{\frac{d-1}{2}}(n)$.

When computational resources, communication power is limited, or the precision of the convex hull algorithm is unnecessary, a lightweight alternative is desirable. In the

following section we develop a stopping algorithm to address these potential feasibility issues.

B. Norm Based Finite-Time Termination

Here we provide a lightweight alternative to the convex hull algorithm, which distributedly finds a norm ball containing all agents' states.

Theorem 4.4: Let $\{c^i(k)\}$ be the value of the i -th agent at time k in a time homogeneous, convex-decreasing protocol. For all $i \in \mathcal{V}$, let

$$R_i(k+1, k') := \max_{j \in N_i^-} \{\|c^i(k'+k+1) - c^j(k'+k)\| + R_j(k, k')\} \quad (4)$$

with $R_i(0, k') := 0$ and $k' \geq 0$. Then

$$c^j(k') \in B\{R_i(D, k'), c^i(k'+D)\}, \quad (5)$$

for all $j \in \mathcal{V}$, where $B\{R, x\}$ denotes the closed ball of radius R centered at x and D is the diameter of the underlying graph topology.

$R_i(k, k')$ is an iterative radius estimate for the process at node i , initialized at time k' after k iterations. The content of the theorem is that after D iterations, the i -th agent has knowledge of a norm ball guaranteed to contain the state of every agent. Note that $R_i(k+1, k')$ is locally computed from $R_j(k, k')$, $c^j(k'+k)$ with j a neighbor to i . To run the norm based stopping each agent i needs only communicate an additional scalar $R_i(k, k')$ to their neighbors alongside their algorithm updates.

Proof: We prove a stronger claim:

$$c^j(k') \in B\{R_i(k, k'), c^i(k'+k)\} \quad (6)$$

for all $j \in N_i^-(k)$. One recovers (5) when $k = D$, as (6) is valid for all $j \in \mathcal{V}$. We prove the claim using induction. For $k = 1$,

$$R_i(1, k') = \max_{j \in N_i^-} \|c^i(k'+1) - c^j(k')\|.$$

Thus for all $j \in N_i^-$, we have $\|c^i(k'+1) - c^j(k')\| \leq R_i(1, k')$ and hence $c^j(k') \in B\{R_i(1, k'), c^i(k'+1)\}$. Thus the assertion holds for $k = 1$.

Now let's assume (6) is true for k . Let $j \in N_i^-(k+1)$, and take q to be an element of a shortest path from j to i , such that $j \in N_q^-(k)$ and $q \in N_i^-$. Then from induction assumption,

$$c^j(k') \in B\{R_q(k, k'), c^q(k+k')\},$$

that is,

$$\|c^q(k'+k) - c^j(k')\| \leq R_q(k, k'). \quad (7)$$

From definition of $R_i(k+1, k')$,

$$\|c^i(k'+k+1) - c^q(k'+k)\| + R_q(k, k') \leq R_i(k+1, k').$$

From triangle inequality,

$$\begin{aligned} \|c^i(k'+k+1) - c^j(k')\| &\leq \|c^i(k'+k+1) - c^q(k'+k)\| \\ &\quad + \|c^q(k'+k) - c^j(k')\|. \end{aligned}$$

Using (7),

$$\begin{aligned} \|c^i(k'+k+1) - c^j(k')\| &\leq \|c^i(k'+k+1) - c^q(k'+k)\| \\ &\quad + R_q(k, k') \end{aligned}$$

which implies that $\|c^i(k'+k+1) - c^j(k')\| \leq R_i(k+1, k')$ and thus, $c^j(k') \in B\{R_i(k+1, k'), c^i(k'+k+1)\}$, and the result follows. ■

Theorem 4.4 provides a distributed way to find a ball which encloses all the nodes. Only information needed by a node is the current radius of its neighbors (along with the states pertaining to ratio consensus) and it can determine the final radius within D iterations. Further, since the ball $B\{R_i(D, k'), c^i(k'+D)\}$ encloses all the nodes, it also encloses the minimum ball, as mentioned earlier. Thus we have provided an algorithm to find an approximation of the minimum ball comprising of all nodes.

Thus if $R_i(D, k)$ is within a tolerance $\rho/2$, all the agents ratio state will be within ρ of consensus. We next provide convergence result for $R_i(D, k)$ as $k \rightarrow \infty$.

Theorem 4.5: For a distributed convex decreasing consensus algorithm c and update as in (4). Let $\bar{R}_i(l) := R_i(D, lD)$ for $l = 0, 1, 2, \dots$ and all $i \in \mathcal{V}$. Then

$$\lim_{l \rightarrow \infty} \bar{R}_i(l) = 0$$

for all $i \in \mathcal{V}$.

We will outline the argument, the interested reader can find a detailed proof in [1]. The idea of the proof is simple, and follows from convergence to consensus. By the assumption that the algorithm achieves consensus the distance between all nodes M is eventually as small as desired. Moreover since $R_i(k+1, lD) \leq M + R_i(k, lD)$, it will follow that $\bar{R}_i(l) \leq MD$, from which the result is obvious since M tends to zero.

Notice that $R_i(l)$ can be different for different nodes and each node might detect ρ -convergence ($R_i(l) < \rho$) at different time instants. According to Theorem 4.4, once $R_i(l) < \rho$ for any $i \in \mathcal{V}$, $\|c^i(lD) - c^j(lD)\| < 2\rho$, that is the ratio state is within 2ρ of consensus value, and the consensus is achieved. Further, any node i which detects convergence can propagate a "converged flag" in the network. To take that into account, we run a separate 1-bit consensus algorithm (denoted as convergence consensus) for each node where each node maintains a convergence state $b_i(k)$ and shares it with neighbors. Each node initializes $b_i(k)$ at every lD iteration for $l \in \{0, 1, 2, \dots\}$ with 1 or 0 depending on the node has detected convergence or not, and updates its value on every iteration using,

$$b_i(k+1) = \bigcup_{j \in N_i^-} b_j(k), \quad (8)$$

where $\bigcup$ denotes OR operation, $k \geq 0$ and $b_j(0) = 1$ if node j has detected convergence at initialization instant 0 and $b_j(0) = 0$ otherwise. Clearly, if $b_j(0) = 1$ for any $j \in \mathcal{V}$, then $b_i(D) = 1$ for all $i \in \mathcal{V}$ where D is the diameter. Thus each node can use $b_i(D)$ as a stopping criterion.

Using above discussion and Theorem 4.5, we present an algorithm (see Algorithm 1) instantiating the result for ratio consensus (which could easily be adapted for more general

settings), which determines the radius $\overline{R}_i(l)$ for $l = 0, 1, 2, \dots$ and all $i \in \mathcal{V}$ and provides a finite-time stopping criterion for vector consensus.

Theorem 4.6: Algorithm 1 converges in finite-time simultaneously at each node.

Proof: From Theorem 4.5, it follows that $\overline{R}_i(l) \rightarrow 0$ as $l \rightarrow \infty$. Thus, for any given $\rho > 0$ and node $i \in \mathcal{V}$ there exists an integer $t(\rho, i)$ such that for $l = t(\rho, i)$, $\overline{R}_i(l) < \rho$. As each node has access to $\overline{R}_i(l)$, convergence can be detected by each node and the convergence bit $b_i(lD + 1)$ will be set to 1. Thus $b_i(lD + D + 1) = 1$ for all $i \in \mathcal{V}$ and algorithm will stop simultaneously at each node. ■

Input:

| $\rho, x^i(0)$; // Initial condition

Initialize:

| $k := 0; R_i(0) := 0; y_i(0) = 1; b_i(0) = 0; l := 1;$

Repeat:

Input:

| $x^j(k), y_j(k), R_j(k), j \in N_i^-$

/* ratio consensus updates of node i given as in Theorem 3.1 */

$x^i(k+1) := \sum_{j \in N_i^-} p_{ji}(k) x^j(k);$

$y_i(k+1) := \sum_{j \in N_i^-} p_{ji}(k) y_j(k);$

$r^i(k+1) := \frac{1}{y_i(k+1)} x^i(k+1);$

/* radius updates of node i given by (4) */

$R_i(k+1) := \max_{j \in N_i^-} \{\|r^i(k+1) - r^j(k)\| + R_j(k)\}$

/* convergence bit update of node i given by (8) */

$b_i(k+1) = \bigcup_{j \in N_i^-} b_j(k,)$

if $k = lD$ **then**

if $b_i(k+1) = 1$ **then**

break; // stop x^i, y_i, r^i, R_i and b_i updates

else

$\overline{R}_i(l) = R_i(k+1);$

if $(\overline{R}_i(l) < \rho)$ **then**

$b_i(k+1) = 1;$ // set convergence bit to 1

else

$R_i(k+1) = 0; b_i(k+1) = 0;$

$l = l + 1;$

end

$k = k + 1;$

Algorithm 1: Finite-time termination of ratio consensus in higher-dimension d (at each node $i \in \mathcal{V}$)

Note, the only global parameter needed for Algorithm 1 is an upper bound on the diameter D , readily available in most applications. Further, the only additional communication required between nodes to implement the algorithm is dimension free, passing a scalar and a single bit for convergence consensus. In comparison, the stopping criterion in

[16] requires $2d$ scalars to be communicated by each node at each iteration. Therefore the extra bandwidth required for each neighbor-neighbor interaction is $B + 1$ where B is the bit length (usually 32) for floating point representation. Thus, the above protocol is suitable for *ad-hoc* communication networks where communication cost is high and bandwidth is limited.

Thus for the applications with high dimensional vector consensus (like GANs, see for example [36]), the algorithm reported here provides a reliable distributed stopping criterion with significantly less communication bandwidth.

We note that the following observation, whose proof follows from an analogous argument given above, that the stopping algorithms introduced here, can also provide bounds on ultimately uniform bounded processes, would not have been possible without the comments of an anonymous reviewer.

Assume the state of our system at time t can be described as $x(t) = (x_1(t), \dots, x_n(t))$ with $x_i(t) \in \mathbb{R}^d$.

Definition 10: $x(t)$ is UUB with ultimate bound b if there exists $c > 0$ such that for $a \in (0, c)$, there exists $T := T(a)$ and $b := b(a)$ such that

$$\|x(t_0)\| \leq a \Rightarrow \|x(t)\| \leq b,$$

for all $t \geq t_0 + T$.

Theorem 4.7: For $x(t)$ UUB and $x_i(t)$ such that $a := R_i(k+1, k') + \|x_i(t)\| < c$, then

$$\|x_j(t')\| \leq b$$

for all $t' \geq t + T$ and $1 \leq j \leq n$.

V. SIMULATION RESULTS

Here we present simulation results to demonstrate finite-time stopping criterion. A network of 25 nodes is considered which is represented by a randomly generated directed graph (see Fig. 2(a)) with diameter 6. The states are 10-dimensional vector and randomly selected randomly. Ratio consensus updates are implemented in MATLAB and simulated. 2-norm of each node's ratio state is plotted in Figure 2(b).

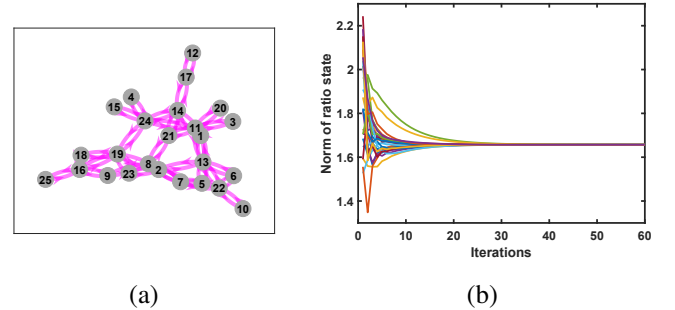


Fig. 2. (a) A communication network represented by a 25 node directed graph. (b) 2-norm of 10-dimensional ratio states of all the nodes (25) in the network.

Algorithm 1 is implemented in MATLAB and the radius $\overline{R}_i(l)$ for all $i \in \mathcal{V}$ is plotted in Fig. 3(a). Here, it can be seen that radius comes under some pre-specified tolerance (0.0166, 1% of the norm of the consensus vector) within 60 iteration and is used as a stopping criterion by each node. Fig. 3(b) plots the two dimensional projection of the *norm* ball

$B\{\bar{R}_i(l), r^i(ID + D)\}$ for node 1 as l progresses over time. As expected, with increase in l , balls shrink in size. Similar observation is seen for all the other nodes as well. The above illustration demonstrates how Algorithm 1 can be successfully used as finite-time termination criterion for distributed ratio consensus.

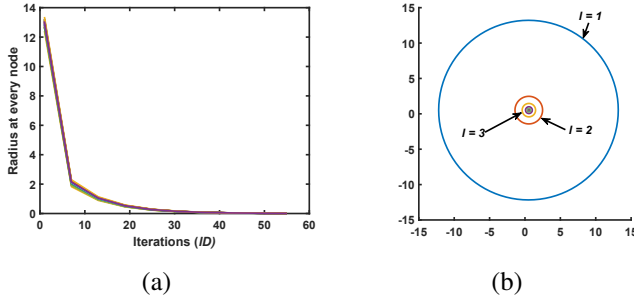


Fig. 3. (a) Radius $\bar{R}_i(l)$ at each node. (b) 2-dimensional projection of norm balls for node 1 with changing l .

VI. CONCLUSION

We presented two algorithms for distributed diameter estimation and a notion of monotonicity of network states, which we called a convex decreasing. In the context of consensus algorithms, We showed that these properties can be combined to construct finite-time stopping criteria. The first algorithm is a distributed convex hull algorithm of independent interest, and the second calculates norm balls which contain all the network states at a given iteration. This algorithm was shown to have much smaller communication requirement compared to existing methods. We thank three anonymous reviewers for careful reading and thoughtful comments which have improved the quality and exposition of this article.

REFERENCES

- [1] J. Melbourne, G. Saraswat, V. Khatana, S. Patel, and M. Salapaka, "Convex decreasing algorithms: Distributed synthesis and finite-time termination in higher dimension," *arXiv:2007.13050*, 2020.
- [2] K. J. Arrow and L. Hurwicz, *Decentralization and computation in resource allocation*. Stanford, Dept. of Economics, 1958.
- [3] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.
- [4] N. A. Lynch, *Distributed algorithms*. Elsevier, 1996.
- [5] J. N. Tsitsiklis, "Problems in decentralized decision making and computation," tech. rep., DTIC Document, 1984.
- [6] C. N. Hadjicostis, A. D. Domínguez-García, and T. Charalambous, *Distributed averaging and balancing in network systems*. Now Foundations and Trends, 2018.
- [7] F. Garin and L. Schenato, "A survey on distributed estimation and control applications using linear consensus algorithms," in *Networked control systems*, pp. 75–107, Springer, 2010.
- [8] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pp. 482–491, IEEE, 2003.
- [9] A. D. Domínguez-García and C. N. Hadjicostis, "Coordination and control of distributed energy resources for provision of ancillary services," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pp. 537–542, IEEE, 2010.
- [10] C. N. Hadjicostis and T. Charalambous, "Average consensus in the presence of delays in directed graph topologies," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 763–768, 2013.
- [11] K. Cai and H. Ishii, "Average consensus on general strongly connected digraphs," *Automatica*, vol. 48, no. 11, pp. 2750–2761, 2012.
- [12] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "A collaborative training algorithm for distributed learning," *IEEE Transactions on Information Theory*, vol. 55, no. 4, pp. 1856–1871, 2009.
- [13] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 115–120, 2002.
- [14] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [15] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 601–615, 2014.
- [16] V. Khatana, G. Saraswat, S. Patel, and M. V. Salapaka, "Gradient-consensus method for distributed optimization in directed multi-agent networks," *2020 American Control Conference (ACC)*, pp. 4689–4694, 2020.
- [17] U. A. Khan, S. Kar, and J. M. Moura, "Distributed sensor localization in random environments using minimal number of anchor nodes," *IEEE Transactions on Signal Processing*, vol. 57, no. 5, pp. 2000–2016, 2009.
- [18] U. A. Khan, S. Kar, and J. M. Moura, "Higher dimensional consensus: Learning in large-scale networks," *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2836–2849, 2010.
- [19] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on automatic control*, vol. 50, no. 2, pp. 169–182, 2005.
- [20] T. Charalambous and C. N. Hadjicostis, "Average consensus in the presence of dynamically changing directed topologies and time delays," in *53rd IEEE Conference on Decision and Control*, pp. 709–714, IEEE, 2014.
- [21] J. Melbourne, G. Saraswat, V. Khatana, S. Patel, and M. V. Salapaka, "On the geometry of consensus algorithms with application to distributed termination in higher dimension," *the proceedings of International Federation of Automatic Control (IFAC)*, 2020.
- [22] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1289–1298, 2006.
- [23] V. Yadav and M. V. Salapaka, "Distributed protocol for determining when averaging consensus is reached," in *45th Annual Allerton Conf*, pp. 715–720, 2007.
- [24] M. Prakash, S. Talukdar, S. Attree, S. Patel, and M. V. Salapaka, "Distributed Stopping Criterion for Ratio Consensus," in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 131–135, Oct. 2018.
- [25] G. Saraswat, V. Khatana, S. Patel, and M. V. Salapaka, "Distributed finite-time termination for consensus algorithm in switching topologies," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 1, pp. 489–499, 2022.
- [26] M. Prakash, S. Talukdar, S. Attree, V. Yadav, and M. V. Salapaka, "Distributed stopping criterion for consensus in the presence of delays," *IEEE Transactions on Control of Network Systems*, 2019.
- [27] S. Sundaram and C. N. Hadjicostis, "Finite-time distributed consensus in graphs with time-invariant topologies," in *2007 American Control Conference*, pp. 711–716, IEEE, 2007.
- [28] R. Diestel, *Graph Theory*. Berlin, Germany: Springer-Verlag, 2006.
- [29] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.
- [30] R. T. Rockafellar, *Convex analysis*. No. 28, Princeton university press, 1970.
- [31] H. Raynaud, "Sur l'enveloppe convexe des nuages de points aléatoires dans $\mathbb{R}^n$," *Journal of Applied Probability*, vol. 7, no. 1, pp. 35–48, 1970.
- [32] I. Bárány and V. Vu, "Central limit theorems for Gaussian polytopes," *The Annals of Probability*, vol. 35, no. 4, pp. 1593–1621, 2007.
- [33] L. Kavan, I. Kolingerova, and J. Zara, "Fast approximation of convex hull," *ACST*, vol. 6, pp. 101–104, 2006.
- [34] D. Fernández-Francos, Ó. Fontenla-Romero, and A. Alonso-Betanzos, "One-class convex hull-based algorithm for classification in distributed environments," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017.
- [35] P. Casale, O. Pujol, and P. Radeva, "Approximate polytope ensemble for one-class classification," *Pattern Recognition*, vol. 47, no. 2, pp. 854–864, 2014.
- [36] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.