Geometric Methods for Resilient Aggregation and Safe Point Computation in Adversarial Multiagent Networks with Imprecise Data

Christopher A. Lee and Waseem Abbas

***Abstract*—This paper studies resilient data aggregation in multiagent networks subject to both adversarial agents and imprecise state observations. We show that existing algorithms, which assume exact state information, fail under such dual uncertainty. To address this, we propose a geometric approach that models each agent's state as an imprecision region (in $\mathbb{R}^d$) containing the true state. We present the *Centerpoint of Imprecision Hulls (CPIH)* algorithm, which takes these regions—some corresponding to adversarial agents—as inputs and computes a point guaranteed to lie within the convex hull of the normal agents' true states, despite unknown adversary identities and true state locations. We thoroughly analyze the algorithm's theoretical guarantees and apply it to the resilient distributed vector consensus problem. Furthermore, we extend the framework to dynamic settings where these regions shrink as agents move closer together, deriving sufficient conditions for exact consensus in a multiagent network despite access to only imprecise states and adversarial presence. Numerical evaluations validate the method's effectiveness.**

I. Introduction

In distributed multiagent networks, agents collaborate to accomplish complex tasks by exchanging information. Their ability to make optimal decisions critically depends on the integrity of this shared data. However, the presence of false or misleading information–stemming from adversarial intent or system failures–can significantly degrade network performance. Ensuring resilience to such abnormal agents is therefore fundamental to the reliable operation of multiagent systems. Extensive research has been devoted to developing resilient data aggregation techniques in the context of distributed optimization, including consensus, estimation, diffusion, learning, and clustering. These efforts have resulted in various algorithms and structural conditions that enhance resilience (e.g., [2]–[11]). The primary objective of these methods is to mitigate or eliminate the influence of adversarial agents, whose identities remain unknown, while ensuring the normal execution of collaborative tasks [12]–[28].

Among the most widely studied approaches are *trimming-based* methods, where the key idea is to discard extreme values and aggregate the remaining ones. A notable example is the family of Mean Subsequence Reduced (MSR)-type algorithms (e.g., [3], [14], [20], [26], [29]), in which each agent collects state values from its neighbors and discards the f largest and smallest values before performing an update. The parameter f is determined by the worst-case number of adversarial agents, the network topology, and the specific distributed task. Another widely adopted approach, particularly for vector-valued states in $\mathbb{R}^d$, exploits the geometric properties of data. Given a set of n points, where up to f points may be adversarial, the goal is to compute a *safe point*—a point that is guaranteed to lie within the convex hull of the normal (non-adversarial) points. Notable techniques in this category include Tverberg partition-based safe point computation (e.g., the ADRC algorithm [12]) and the centerpoint-based aggregation methods [13], [30].

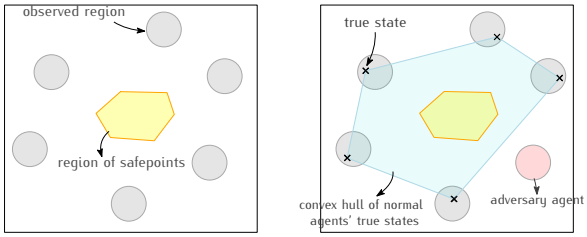
However, existing resilient solutions assume that agents can precisely observe their neighbors' unperturbed states—an assumption rarely met in practice due to sensor noise, environmental fluctuations, and hardware constraints [31]–[35]. We show that such measurement errors can accumulate over time, leading even well-designed resilient algorithms to fail, despite adherence to prescribed network conditions and limits on adversarial agents. This underscores the need for new strategies that *enable agents to resiliently aggregate data in the presence of adversarial neighbors and uncertainty in state observations, even from non-adversarial agents.*

To address this challenge, we propose a resilient data aggregation framework for distributed multiagent systems in which agents observe their neighbors' states with bounded imprecision. Specifically, we model each observed state as deviating from its true value by a displacement vector of bounded magnitude—termed the radius of imprecision—such that an agent perceives regions that contain the true states of its neighbors. This uncertainty is further exacerbated by the presence of adversarial agents, whose identities remain unknown. We present an aggregation algorithm that enables an agent to compute a safe point—a point guaranteed to lie within the convex hull of the true states of normal (non-adversarial) neighbors—despite lacking direct access to these states or the ability to identify adversaries, as illustrated in Figure 1. Our approach builds on centerpoint-based aggregation methods, extending the concept of a centerpoint, which is a generalization of the median in higher-dimensional spaces, to operate on regions rather than discrete points in $\mathbb{R}^d$. We further establish theoretical conditions on the radius of imprecision and the number of adversarial agents to ensure that the safe point exists and remains within the convex hull of the normal neighbors' true states. While applicable to any distributed optimization task requiring resilient aggregation, we demonstrate our method's efficacy through resilient distributed consensus.

When the radius of imprecision in agents' state observations becomes excessively large, the resulting uncertainty may preclude the existence of a safe point, making its computation infeasible using existing methods. To address this, we extend our framework to a dynamic setting where the radius of imprecision decreases over time—a realistic assumption in systems where agents improve state observation precision as

C. Lee is with the Electrical Engineering Department at the University of Texas at Dallas, Richardson, TX. W. Abbas is with the Systems Engineering Department at the University of Texas at Dallas, Richardson, TX, USA (Emails: christopher.lee3@utdallas.edu, waseem.abbas@utdallas.edu).

Some preliminary results appeared in [1].



(a) (b)

Fig. 1: (a) There are six agents, where any one agent may be adversarial. Instead of knowing the agents' true states, we only have access to 'observed regions' containing these true states. Our algorithm computes the safepoint region, ensuring that all points within it lie inside the convex hull of the normal agents' true states. (b) An example where the true states of five normal agents are shown, along with the convex hull of their true states (blue region). Notably, the safe point region (yellow) is entirely contained within the convex hull.

they converge spatially. In resilient distributed consensus, for instance, agents aim to reach agreement on a common point, leading to reduced inter-agent distances and, consequently, decreasing observation uncertainty. We derive sufficient conditions on the decay rate of the radii of imprecision for normal (non-adversarial) agents, ensuring that a safe point exists and remains computable at each time step. As a concrete example, we demonstrate that under these conditions in resilient distributed consensus, all normal agents converge to a common state within the convex hull of their initial states.

We summarize our key contributions as follows:

- We propose a resilient data aggregation framework for multiagent systems where agents observe neighbors' states with bounded imprecision, modeled as regions containing true states. We show that existing resilient aggregation methods (e.g., the Centerpoint based method) fail in this setting, even when the number of adversarial agents is within prescribed limits.
- We propose handling imprecise state observations by replacing point-based representations in $\mathbb{R}^d$ with *imprecision regions*—sets that contain the true states of agents. We further introduce the concept of the *invariant hull*, which represents the largest region guaranteed to be contained within the convex hull of agents' true states despite only knowing imprecise observations. We then provide a geometric characterization of invariant hulls and present an efficient algorithm for their computation.
- We develop a resilient aggregation method—*Centerpoint of Invariant Hulls (CPIH)*—that leverages intersecting subsets of invariant hulls to identify a safe point within a normal agent's neighborhood, even when it contains up to $\frac{N_i}{d+1}-1$ adversarial agents. This safe point enables normal agents to update their states while effectively rejecting adversarial misinformation and compensating for imprecise observations of neighbors' states. As an application, we apply this method to the resilient distributed consensus problem and also perform numerical evaluations.
- We extend our framework to account for *dynamic imprecision*, where the size of imprecision regions decreases as agents move closer together. We derive a sufficient condition on the rate of imprecision decay that guarantees the continued existence of safe points, ensuring that normal agents ultimately achieve exact resilient consensus despite adversarial agents and state imprecision via the *Dynamically bounded CPIH (DB-CPIH)* algorithm. We also illustrate the efficacy through simulations.

The rest of the paper is organized as follows: Section II introduces the preliminaries and notation. Section III reviews existing resilient algorithms and highlights their limitations in the presence of imprecise state observations. Section IV introduces the concept of the invariant hull, discusses its computation, and explains its significance. Section V presents the Centerpoint of Invariant Hulls (CPIH) algorithm for resilient aggregation, while Section VI extends CPIH to handle dynamic imprecision. Finally, Section VII concludes the paper.

II. Preliminaries

We consider an undirected graph $G = (V, E)$, modeling a multiagent network. Here, V is the set of agents and E denotes the interactions between them. An edge connecting agents v_i and v_j is represented by the unordered pair (v_i, v_j). The terms *agent* and *node* are used interchangeably. Each agent $v_i \in V$ possesses a d-dimensional state vector, denoted by $x_i(t) \in \mathbb{R}^d$, that evolves over time t. The *neighborhood* of v_i is the set of nodes $N_i = \{v_j \in V : (v_i, v_j) \in E\} \cup \{v_i\}$ (node v_i is included in its neighborhood). If the underlying communication network may change over time, $N_i(t)$ denotes the neighborhood of v_i at time t. For a given set of points $X \subset \mathbb{R}^d$, its convex hull is denoted by $\texttt{Conv}(X)$. Additionally, we use the terms *points* and *states* interchangeably. The network comprises agents categorized as either *normal* or *adversarial*. Normal agents, denoted by $V_n \subseteq V$, engage in synchronous interactions with their neighbors and update states based on a predefined consensus algorithm. Conversely, adversarial agents, denoted by $V_f \subset V$, can modify their states arbitrarily. An adversarial agent can transmit distinct values to different neighbors, following the *Byzantine* model. Importantly, a normal node cannot discern which neighbors are adversarial.

State Imprecision – We consider a setting where each agent possesses only imprecise information about the states of its neighbors. Specifically, when an agent v_i interacts with a neighboring agent v_j, it observes v_j's state with some degree of imprecision, which may arise from factors such as calibration errors, hardware inaccuracies, or measurement uncertainties. We denote v_i's imprecise observation of agent v_j's state at time t as $r_{i(j)}(t)$. If $x_j(t)$ is the true state of v_j at time t, we assume:

$$\left\| r_{i(j)}(t) - x_j(t) \right\| \leq \delta_j(t),$$

where $\delta_j(t)$ is the *imprecision radius* associated with v_j at time t. This means that agent v_i associates an *imprecision region* with v_j, denoted as

$$B_{i(j)}(t) = \left\{ p \in \mathbb{R}^d : \left\| p - r_{i(j)}(t) \right\| \leq \delta_j(t) \right\},$$

which ensures that the true state of v_j, i.e., $x_j(t)$ must lie within this region. In the fixed imprecision setting, we assume that the imprecision radius for each normal agent remains constant over time, i.e., $\delta_i(t) = \delta$ for all $t > 0$, where δ is a known parameter. We note that our methods, developed across subsequent sections, apply to imprecision regions modeled as arbitary polytopes of uniform orientation in $\mathbb{R}^d$. However, the d-dimensional ball simplifies distance computations due to its symmetry, so we adopt the Euclidean ball (as in Figure 2(b)) or the L_∞ ball (a hypercube, Figure 2(c)) as natural models throughout this paper.

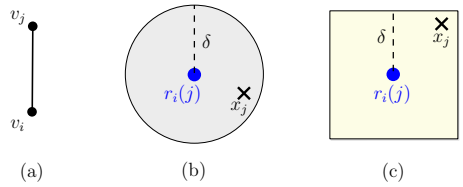


Fig. 2: (a) Agents v_i and v_j are neighbors with true states x_i and x_j in $\mathbb{R}^2$, respectively. (b) Imprecision region associated by v_i with v_j, where $r_i(j)$ represents agent v_i's observation of v_j's state. (c) An alternative imprecision model with a d_∞-ball imprecision region, where the deviation between the observed state and the true state in each dimension is at most δ.

In Section VI, we consider the dynamic imprecision setting, where $\delta_i(t)$ evolves over time. Imprecision regions may generally differ across observing agents, i.e., $B_{i(j)} \neq B_{k(j)}$ for $i \neq k$, reflecting distinct perspectives on v_j's state. However, for clarity, we simplify the notation to B_j whenever the identity of the observing agent is not relevant to the discussion. Table I provides a summary of the notation that will subsequently be explicitly defined.

$x_i(t)$	True state of v_i at time t.
$r_{i(j)}(t)$	Observed state of v_j by v_i at time t.
$\delta_i(t)$	Imprecision radius of v_i at time t.
$R_{N_i}(t)$	Set of observed states of the neighbors of v_i at time t, i.e., $\{r_{i(j)}(t) : v_j \in N_i(t)\}$.
$B_{i(j)}(t)$	Imprecision region associated with agent j, centered at $r_{i(j)}(t)$.
$B_{N_i(t)}$	The set of imprecision regions for the neighbors of agent v_i at time t, i.e., $\{B_{i(j)}(t) : v_j \in N_i(t)\}$.
$P(B_{N_i}(t))$	Potential configuration, consisting of a selection of one point from each imprecision region in $N_i(t)$.
$\mathcal{P}(B_V)$	The set of all potential configurations given a set of imprecision regions B_V.
$\texttt{IHull}(B_{N_i}(t))$	Invariant hull, consisting of points that are in the convex hull of every potential configuration of $B_{N_i}(t)$.

TABLE I: Notation.

To illustrate our methods, we focus on the *resilient vector consensus* problem, which we define below.

Resilient Consensus Problem – In a network containing both normal and adversarial agents, the objective of resilient vector consensus is to design a distributed protocol that ensures all normal agents update their states to eventually converge on a common state. This common state must lie within the convex hull of their initial states, denoted as $X_n(0) = \{x_1(0), x_2(0), \cdots, x_n(0)\}$. The protocol must satisfy the following conditions:

- *Safety:* At any time step t, the state of any normal node v_i must lie within $\texttt{Conv}(X_n(0))$.
- *Agreement:* For every $\epsilon > 0$, there exists a time t_ϵ, such that for any two normal agents v_i and v_j, $\|x_i(t) - x_j(t)\| < \epsilon$ for all $t > t_\epsilon$.

Next, we review existing resilient consensus solutions and discuss how imprecision in agent state information can affect their performance.

III. Effect of Imprecision

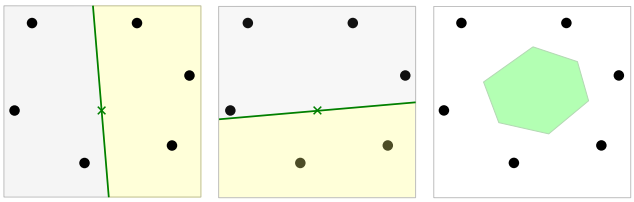
In this section, we demonstrate that existing resilient aggregation methods fail when subjected to imprecision. Specifically, we illustrate this in the context of the resilient vector consensus problem. The centerpoint-based resilient consensus algorithm (CP algorithm) achieves optimal resilience, guaranteeing the convergence of all normal agents to a common point despite the presence of the maximum tolerable number of adversarial agents in each normal agent's neighborhood (as detailed below). We begin with a brief review of the CP algorithm and then show how state imprecision leads to its failure, as well as the failure of other existing imprecision-unaware algorithms, in computing valid safe points, thereby undermining resilient consensus.

A. Resilient Consensus with No Imprecision

A variety of resilient consensus algorithms that have been proposed to accommodate both scalar and multi-dimensional states, e.g., [2], [3], [12]–[23], [28], [29]. In the general case, where states have dimension $(d > 1)$, these algorithms typically follow a common structure: at each step of the consensus process, normal nodes compute a point that lies within the interior of the convex hull of their normal neighbors' states. They then update their states by moving toward this point, referred to as the **safe point**. Computing a safe point is a challenging problem, and various techniques have been developed to achieve this. Among them, the centerpoint-based resilient consensus algorithm (CP algorithm) stands out due to its superior resilience—in terms of the maximum number of tolerable adversarial agents. The CP algorithm computes a centerpoint of the agents' states and uses it as the safe point. The concept of a centerpoint, a high-dimensional generalization of the median, is formally defined as follows.

Definition 3.1. *(Centerpoint) Given a set S, of N points in $\mathbb{R}^d$, a centerpoint p is a point with the property that every closed halfspace of $\mathbb{R}^d$ containing p must also contain at least $\frac{N}{d+1}$ points of S.*

The set of all centerpoints is referred to as the *centerpoint region*. Figure 7 illustrates an example. There are six points in $\mathbb{R}^2$, and any line passing through a centerpoint divides these six points into two regions, each containing at least two points, as in Figures 7(a) and (b). Figure 7(c) illustrates the centerpoint region for the given example.



(a) (b) (c)

Fig. 3: Illustration of centerpoint. In (a) and (b), centerpoint is denoted by '$\times$' and lines are passing through the centerpoint. The green shaded region in (c) is the centerpoint region.

The notion of a centerpoint provides a complete characterization of the safe point. It is shown in [13] that a safe point for an agent v_i is essentially a centerpoint of its neighbors' states, provided that that the number of adversaries in the neighborhood of v_i is bounded by $\frac{|N_i(t)|}{d+1}$. Here, d is the dimension of the state and $|N_i(t)|$ is the number of neighbors of v_i. If the number of adversarial agents exceeds this bound, a safe point may not exist, underscoring the CP algorithm's utility in maximizing resilience.

Now, a *resilient consensus algorithm* based on centerpoints, assuming *no imprecision*—agents observe their normal neighbors' states exactly—can be designed as follows:

- In each iteration t, a normal agent v_i gathers the state values of its neighbors in $N_i(t)$, and computes a safe point $s_i(t)$ by determining a centerpoint of neighbors' states.
- Agent v_i updates its states as follows:

$$x_i(t+1) = \alpha_i(t)s_i(t) + (1-\alpha_i(t))x_i(t), \qquad (1)$$

where $\alpha_i(t) \in (0\ \ 1)$ is a dynamically chosen parameter whose value depends on the application [12].

Next, we examine how the above resilient consensus algorithm, as well as other safe point-based approaches, perform when agent states are subject to bounded imprecision.

B. Failure of Resilient Solutions Under Imprecision

Resilient consensus algorithms rely on agents computing safe points within the convex hull of their non-adversarial neighbors' true states. Under state imprecision, however, these true states are unknown, and methods that ignore imprecision may fail to produce valid safe points. It is important to note that when state values are subject to fixed imprecision, achieving true consensus is, in general, impossible. However, a resilient consensus algorithm should, at minimum, ensure *approximate consensus*—that is, rather than guaranteeing that all normal agents converge to an exact common point, they should converge within a bounded region contained inside the convex hull of their initial states. Unfortunately, existing resilient consensus algorithms may fail to achieve even approximate consensus. For example, a centerpoint computed by a normal agent based on the *observed states* (which are affected by imprecision) of its neighbors does not necessarily lie within the convex hull of the *true states* of its normal neighbors. Consequently, the computed centerpoint is not a valid safe point. Figure 4 illustrates this issue.

In Figure 4(a), we consider a scenario with six agents in the neighborhood of a normal agent v_i (including v_i itself). Among them, the red agent v_f is adversarial, but v_i remains unaware of which agent is adversarial. Each agent's state has an associated imprecision region, which is assumed to be a square. The centerpoint region computed based on observed states (represented by '$\bullet$') is highlighted in green, whereas the convex hull of normal agents' true states (indicated by '$\times$') is depicted as gray. Figure 4(b) highlights the challenge posed by imprecision. The centerpoint region, computed using observed states, fails to remain entirely within the convex hull of the normal agents' true states. Consequently, agent v_i may select a centerpoint (indicated by '$\circ$') that is *not a safe point*. This, in turn, causes v_i to update its state in a direction outside the convex hull of the normal agents' true states, thereby violating the safety condition required for resilient consensus. We note that a similar issue arises with other imprecision region geometries, such as circular regions, further highlighting the pervasive challenge that imprecision poses across different modeling assumptions.

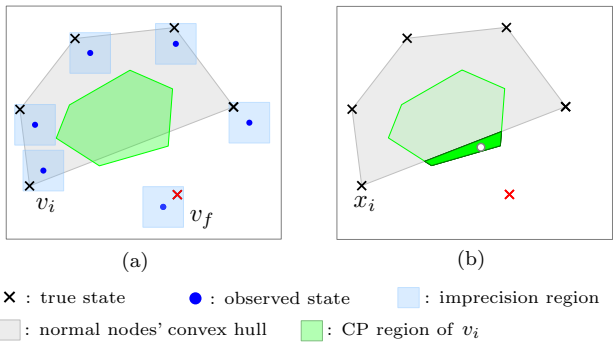
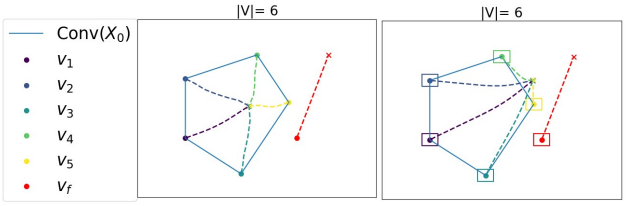


Fig. 4: Centerpoint region based on the observed states is not contained entirely in the convex hull of normal agents' true states.

Figure 5 illustrates the behavior of the resilient consensus algorithm with and without state imprecision. The setup consists of six agents, all of whom are pairwise adjacent, with one adversarial agent. In the absence of imprecision, all normal agents converge within the convex hull of their initial states despite the adversary, as shown by their state trajectories in Figure 5(a). However, under imprecision (modeled as square regions), the resilient consensus algorithm fails. As depicted in Figure 5(b), normal agents do not remain within the convex hull of their initial states and instead continue to drift farther away, demonstrating the disruptive impact of imprecision.

The primary challenge stems from the *inability to reliably compute a safe point when observed states are imprecise*. This necessitates the development of new approaches to ensure the



(a) (b)

Fig. 5: (a) Normal agents achieve resilient consensus with no imprecision. (b) Normal agents do not converge within the convex hull of normal agents' initial states due to imprecision.

accurate computation of safe points despite imprecision. In the next section, we address this problem.

IV. Resilience in the Presence of Imprecision

In this section, we develop an aggregation method that enables normal agents to *compute safe points even when encountering imprecise state values from their neighbors*.

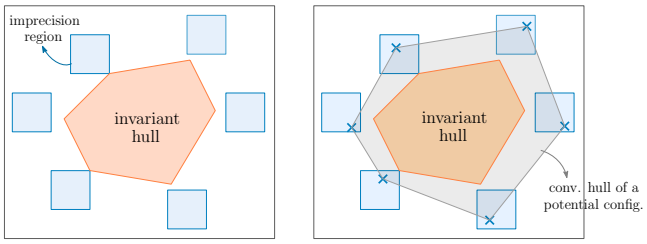
Formally, the problem we address is as follows:

Resilient Consensus with Fixed Imprecision – For a network containing both normal and adversarial agents, assume that each normal agent observes imprecise estimates of its neighbors' states, such that the true state vectors are known to lie within a fixed error bound around these observations. The objective is to design a distributed protocol that ensures all normal agents update their states in a way that guarantees convergence to a final value within the convex hull of the initial states of the normal agents, despite the presence of adversarial agents and observation imprecision. Specifically, the protocol must satisfy the following conditions:

- *Approximate Agreement:*
  1) For all $t > 0$, $\text{Conv}(X_n(t)) \subseteq \text{Conv}(X_n(0))$.
  2) The protocol must reach a final state $X_n^* = X(t^*) \subset X_n(0)$ for some $t^* > 0$, such that for all $t' > t^*$ $X_n(t') = X_n^*$.

Next, we will discuss existing resilient consensus solutions and the impact that imprecision of agent states may have on their performance. When a normal agent v_i observes an imprecise state of its neighbor v_j, the agent v_i essentially perceives an imprecision region associated with v_j containing the true state value of v_j. Thus, v_i effectively observes a collection of such imprecision regions corresponding to its neighbors. By selecting one value (a point) from each imprecision region, we define a *potential configuration* of the agents' states. Notably, the true states of the agents represent only one among infinitely many possible potential configurations. Our goal is to identify the largest region—termed the *invariant hull*—contained within the convex hull of *every* possible potential configuration of agents' states. By construction, the invariant hull is always a subset of the convex hull of the agents' true states. At a high level, the invariant hull for a given set of imprecision regions is akin to the convex hull of a given set of points. Figure 6 illustrates these concepts.

Figure 6(a) shows a set of imprecision regions (blue boxes) and the associated invariant hull. Figure 6(b) presents one potential configuration (set of points in imprecision regions indicated by '×') and the associated convex hull of the potential configuration (gray shaded region). Notably, the invariant hull is a subset of the convex hull of the selected potential configuration shown. This property holds for any potential configuration—regardless of how points are selected from the imprecision regions, the invariant hull will always be contained within the convex hull of that configuration.



(a) (b)

Fig. 6: (a) Invariant hull of a set of imprecision regions. (b) Invariant hull is a subset of the convex hull of an arbitrary potential configuration.

We now summarize some notation and formally define *invariant hull*. For convenience, we will omit the time dependence (t) in notation when referring to sets of regions, states, or neighborhoods at an arbitrary time instance, provided the context is clear. The following definitions and results focus on the geometric properties of imprecision regions as subsets of $\mathbb{R}^d$, independent of any specific agent v_i. To facilitate this discussion, we denote a generic set of imprecision regions by B_V, allowing us to analyze invariant hull properties in a general setting without explicitly referencing B_{N_i} within the multiagent framework.

Definition 4.1. *(Potential Configuration) For a given set of imprecision regions $B_V = \{B_1, \cdots, B_n\}$, a potential configuration is a set of points $P(B_V) = \{p_1, \cdots, p_n\}$ such that $p_i \in B_i$ for each $v_i \in V$.*

Now, we define the notion of an invariant hull.

Definition 4.2. *(Invariant Hull) Consider a set of imprecision regions B_V. Let $\mathcal{P}(B_V)$ be the set of all possible potential configurations, and $\texttt{Conv}(P)$ denotes the convex hull of a potential configuration $P \in \mathcal{P}(B_V)$. Then, the invariant hull of B_V is defined as,*

$$\texttt{IHull}(B_V) = \bigcap_{P \in \mathcal{P}(B_V)} \texttt{Conv}(P). \tag{2}$$

In simpler terms, for the invariant hull, we find all possible potential configurations of B_V, compute the convex hull of each such configuration, and finally compute the intersection of all such convex hulls. Consequently, the invariant hull is essentially a subset of the convex hull of all potential configurations of B_V, as Figure 6 illustrates. Moreover, the invariant hull is always a convex set.

Next, we provide a geometric characterization of the invariant hull. Then, in Section IV-B, we present an efficient method to compute the invariant hull of B_V.

A. Characterization of Invariant Hull

The arguments to follow deal extensively with combinatorial families of sets, thus we introduce the following notation for convenience:

- Consider a set of imprecision regions B_V, *then* B_V^{d+1} *denotes a family of all* $(d+1)$*-member subsets of* B_V (i.e., subsets of B_V consisting of $d+1$ imprecision regions). For example, consider $V = \{v_1, v_2, v_3, v_4\}$, the corresponding $B_V = \{B_1, B_2, B_3, B_4\}$, and $d = 2$, then
$$B_V^3 = \{\ \{B_1, B_2, B_3\},\ \{B_1, B_2, B_4\}\ \{B_1, B_3, B_4\},\ \{B_2, B_3, B_4\}\ \}.$$
- Similarly, for $Q \in B_V^{d+1}$, the notation Q^d denotes the family of all d-member subsets of Q. In the above example, if $Q = \{B_1, B_2, B_3\} \in B_V^3$, then
$$Q^2 = \{\{B_1, B_2\},\ \{B_1, B_3\},\ \{B_2, B_3\}\ \}.$$

So, in general, a *superscript* on a set notation denotes the combinatorial family of the set. In the following arguments, we will use the notation h^+ and h^- to denote opposite closed halfspaces bounded by a hyperplane h. Whereas conventionally, h^-/h^+ would denote the sets of points below/above h, we do not require this specification in the context to follow. Additionally, we assume that B_V is a set of mutually disjoint imprecision regions and that imprecision is fixed such that $\underset{B_i \in B_V}{\forall} \delta_i = \delta$, for some constant δ. The following lemma identifies an essential property that all points in the invariant hull must possess.

Lemma 4.1. *Let* $B_V = \{B_1, .., B_{d+1}\}$ *be a* $(d+1)$*-member subset of imprecision regions in* $\mathbb{R}^d$*. If* $\texttt{IHull}(B_V)$ *is non-empty, then a point* $p \in \texttt{IHull}(B_V)$ *if and only if* p *has the following property:*

Property 1. *For every hyperplane* h *passing through* p*, at least one imprecision region* $B_i \subset B_V$ *is contained in each of the associated closed halfspaces of* h*.*

Proof. We prove the contrapositive. If $p \notin \texttt{IHull}(B_V)$, then by Definition 4.2 there exists at least one potential configuration $P(B_V) = (x_1, x_2, \cdots, x_{d+1}) \in \mathcal{P}(B_V)$ such that $p \cap \texttt{Conv}(P(B_V)) = \emptyset$. Since p and $\texttt{Conv}(P(B_V))$ are disjoint, it follows that there is a hyperplane h, intersecting p such that $P(B_V)$ is contained in one of its halfspaces. Let h^- denote the open halfspace bounded by h that contains $P(B_V)$ and h^+ denote the opposite open halfspace. Since each of the $d+1$ points of $P(B_V)$ are selected from the $d+1$ imprecision regions of B_V, it follows that $\underset{B_i \in Q}{\forall} B_i \cap h^- \neq \emptyset$, and therefore no imprecision region is a subset of h^+. Thus, if $p \notin \texttt{IHull}(B_V)$, then p does not have Property 1.

Similarly, if a point p does not possess Property 1, then there is a hyperplane h' intersecting p such that no imprecision region of B_V is a subset of h'^+. Then it is possible to select $P(B_V) \in \mathcal{P}(Q)$ such that $P(B_V) \subset h'^-$. Since $p \cap h'^- = \emptyset$, p and $P(B_V)$ are disjoint, which implies that $p \notin \bigcap_{P \in \mathcal{P}(B_V)} \texttt{Conv}(P)$ and therefore $p \notin \texttt{IHull}(B_V)$ by Definition 4.2. ∎

We now present an auxiliary lemma that will be used in proving Theorem 4.4 characterizing the invariant hulls.

Lemma 4.2. *Let* B_V *be a set of* $d+1$ *imprecision regions in* $\mathbb{R}^d$*, and let* p *be a point contained in* $int(\texttt{Conv}(B_V))$*. Then* $p \in \texttt{IHull}(B_V) \iff \underset{q \in B_V^d}{\forall} p \notin int(\texttt{Conv}(q))$.

Note: Here and throughout the paper, if K *is a family of point sets, then* $\texttt{Conv}(K) = \texttt{Conv}(\bigcup_{k \in K} K)$*. Additionally, "*$int$*" refers to the interior.*

Proof. To prove the forward implication, assume $p \in \texttt{IHull}(B_V)$. Now, we observe that if $p \in int(\texttt{Conv}(q))$, for any $q = \{B_1, B_2, \cdots, B_d\} \in B_V^d$, then $p = a_1 x_1 + a_2 x_2 + \cdots + a_d x_d$, where $\sum_{i=1}^{d} a_i = 1$, and $\underset{1 \le i \le d}{\forall} x_i \in int(B_i)$. It follows that the hyperplane h that intersects each of $x_1, x_2, \cdots, x_d$ must also intersect p. Note that the existence of h is assured, since there is a hyperplane through any d points in $\mathbb{R}^d$. Since h intersects the interior of each $B_1, B_2, \cdots, B_d$, none of the d members of q are contained in either h^+ or h^-, which implies that at most one $B_i \in B_V \setminus q$ can be contained in one of h^+ or h^-. Thus one of either h^+ or h^- contains no $B_i \in B_V$ as a subset. Consequently, p does not possess Property 4.1 and by Lemma 4.1, $p \notin \texttt{IHull}(B_V)$. We conclude that $p \in \texttt{IHull}(B_V) \rightarrow \underset{q \in B_V^d}{\forall} p \notin int(\texttt{Conv}(q))$. For the reverse implication, we assume $\underset{q \in B_V^d}{\forall} p \notin int(\texttt{Conv}(q))$. Since $p \in int(\texttt{Conv}(B_V))$, there exists a configuration $P(B_V) = \{(x_1, x_2, \cdots, x_{d+1}) : \underset{i}{\forall} x_i' \in int(B_i)\}$ such that $p \in \texttt{Conv}(P(B_V))$. For the sake of contradiction, assume that $p \notin \texttt{IHull}(B_V)$. Then it is possible to select a different configuration, $P'(B_V) = \{(x_1', x_2', \cdots, x_{d+1}') : \underset{1 \le i \le d+1}{\forall} x_i' \in int(B_i)\}$, such that $p \notin \texttt{Conv}(P'(B_V))$. Each d-dimensional face of $\texttt{Conv}(P'(B_V))$ is formed by $\texttt{Conv}(x_1' \cup x_2' \cup \cdots \cup x_d')$ for some d-tuple of points of $P'(B_V)$. Since $p \in P(B_V)$, but $p \notin P'(B_V)$, then for some $q = \{B_1, B_2, \cdots, B_d\} \in B_V^d$, $p \in int(\texttt{Conv}\left(f_q \cup f_q'\right))$, where $f_q = \texttt{Conv}(\bigcup_{x_i \in q \cap P(B_V)} x_i)$ and $f_q' = \texttt{Conv}(\bigcup_{x_i \in q \cap P'(B_V)} x_i)$ (i.e., f_q is a face of $\texttt{Conv}(P(B_V))$, and f_q' is the corresponding face of $\texttt{Conv}(P'(B_V))$). But $int(f_q \cup f_q') \subset int(\texttt{Conv}(q))$, so $p \in int(f_q \cup f_q')$ contradicts our assumptions. Therefore $\underset{q \in B_V^d}{\forall} p \notin int(\texttt{Conv}(q)) \rightarrow p \in \texttt{IHull}(B_V)$. ∎

We now show that the invariant hull of n imprecision regions is the convex hull of the union of invariant hulls of each of the $\binom{n}{d+1}$ subsets of B_V^{d+1}. The proof will make use of the following generalization of the theorem of Caratheodory to sets in $\mathbb{R}^d$:

Theorem 4.3. *[36] For a family of sets $K \in \mathbb{R}^d$, with $|K| \geq d+1$, $\mathtt{Conv}(K) = \bigcup_{k_i \in K^{d+1}} \mathtt{Conv}(k_i)$.*

In other words the convex hull of every point contained in the family of sets is equivalent to the union of the convex hulls of all of its $(d+1)$-member subsets. Now, we state a key result that provides a direct means of computing the invariant hull.

Theorem 4.4. *Let $B_V = \{B_1, \cdots, B_n\}$ be a collection of n imprecision regions in $\mathbb{R}^d$, and B_V^{d+1} denote the family of $(d+1)$-member subsets of B_V. Then,*

$$\mathit{IHull}(B_V) = \mathit{Conv}\left(\bigcup_{Q \in B_V^{d+1}} \mathit{IHull}(Q)\right) \tag{3}$$

Proof. First, note that $\mathtt{IHull}(B_V)$ is defined in Definition 4.2 to be the intersection of infinite convex sets. Thus, $\mathtt{IHull}(B_V)$ is also a convex set of $\mathbb{R}^d$ since convexity is preserved under intersection. Let z be an arbitrary vertex of $\mathtt{IHull}(B_V)$. Clearly, $z \in \mathtt{Conv}(B_V)$. It follows from Theorem 4.3 that

$$\exists Q_i : Q_i \in B_V^{d+1} \wedge z \in \mathtt{Conv}(Q_i)$$

For the following argument, let $f \in \mathcal{F}(P(B_V))$ denote a d-dimensional face of the convex hull of a given configuration $P(B_V)$, and $\mathcal{F}(P(B_V))$ denote the set of all faces of $\mathtt{Conv}(P(B_V))$. Now, if z is a vertex of $\mathtt{IHull}(B_V)$, then $z = f_1 \cap f_2 \cap \cdots \cap f_d$ for some $\{f_1, f_2, \cdots, f_d\} : f_i \in \mathcal{F}(P_i(B_V))$. Note that $f_i = \mathtt{Conv}(x_{i_1} \cup x_{i_2} \cup \cdots \cup x_{i_d})$, where each $x_{i_j} \in B_j$ for one $B_j \in B_V$, and for $B_j \neq B_k$, $B_j \cap B_k = \emptyset$. It follows that each f_i contains a point x_{ij} contained in a single $B_j \in B_V$. This implies that $z \in B_j$ and $z \notin int(B_j)$, since for any $p \in int(B_j)$, there exists a point $p' \in \partial B_j$, and a configuration $P'(B_V)$ with faces $f'_1, f'_2, \cdots, f'_d \in \mathcal{F}(P'(B_V))$ such that $f'_1 \cap f'_2 \cap \cdots \cap f'_d = p'$ and $p \cap \mathtt{Conv}(f'_1 \cup f'_2 \cup \cdots \cup f'_d) = \emptyset$, which implies $p \notin \mathtt{Conv}(P'(B_V))$ and therefore $p \notin \mathtt{IHull}(B_V)$. To summarize, z is the intersection of d faces, $f_1, f_2, \cdots, f_d$, each of which is the convex hull of d points, $\{x_{i_1}, x_{i_2}, \cdots, x_{i_d}\}$ from a set of d imprecision regions $B_F = \{\bigcup_{i=1}^{d} B_{i_1} \cup B_{i_2} \cup \cdots \cup B_{i_d} : x_{i_k} \in B_k, B_j \in B_F\}$. From Theorem 4.3, we have that $z \in \mathtt{Conv}(Q)$, for some $Q \in \mathtt{Conv}(B_F^{d+1})$. Furthermore, $z \notin int(B_j) \rightarrow z \notin int(\mathtt{Conv}(Q^d)$, since every $d+1$ region subset of B_F includes B_j. Therefore, Lemma 4.2 implies that $z \in \mathtt{IHull}(Q)$. Since this holds for any vertex $z \in \mathtt{IHull}(B_V)$, and for each $\mathtt{IHull}(Q)$, $\mathtt{IHull}(Q) \subset \mathtt{IHull}(B_V)$, this concludes the proof. ∎

Corollary 4.5. *Let B_V be a set of n imprecision regions in $\mathbb{R}^d$. If $n = d+1$, then*

$$\mathit{IHull}(B_V) = \mathit{Conv}(B_V) \setminus int(\mathit{Conv}(\bigcup_{q_i \in B_V^d} q_i).$$

If $n \geq d+1$, then

$$\mathit{IHull}(B_V) = \mathit{Conv}(\mathit{Conv}(B_V) \setminus int(\mathit{Conv}(\bigcup_{q_i \in B_V^d} q_i))).$$

Here we have used the set notation of "$A \setminus B$" to refer to the set of all elements of A that are not elements of B.

Proof. For the case of $n = d+1$, $\mathtt{IHull}(B_V) = \mathtt{Conv}(B_V) \setminus int(\mathtt{Conv}(\bigcup_{q_i \in B_V^d} q_i)$ is a direct implication of Lemma 4.2. So for the case of $n > d+1$, given any $Q \in B_V^{d+1}$, we have that $\mathtt{IHull}(Q) = \mathtt{Conv}(Q) \setminus int(\mathtt{Conv}(\bigcup_{q_i \in Q^d} q_i))$. From Theorem 4.4, $\mathtt{IHull}(B_V) = \mathtt{Conv}\left(\bigcup_{Q \in B_V^{d+1}} \mathtt{IHull}(Q)\right) = \mathtt{Conv}\left(\bigcup_{Q \in B_V^{d+1}} \mathtt{Conv}(Q) \setminus int(\mathtt{Conv}(\bigcup_{q_i \in Q^d} q_i)\right)$. Applying the relation $\bigcup_{Q \in B_V^{d+1}} \mathtt{Conv}(Q) = B_V$, from Theorem 4.3, we arrive at the expression given in Corollary 4.5. ∎

B. Computing the Invariant Hull

We now outline a procedure for computing the invariant hull of B_V, where $|B_V| \geq d+1$. The procedure involves computing the equation of a tangent hyperplane to every set of d imprecision regions in $\mathbb{R}^d$. In the case where imprecision regions are polygons in $\mathbb{R}^2$, linear-time algorithms have been developed to find *outer* tangent lines, that is, the tangent lines that have all participating polygons on one side [37]. For the case in which imprecision regions are modeled as d-balls, the spherical boundaries can be expressed as $x \top A x = 0$, where x is in homogeneous coordinates, and A is an invertible $(d+1) \times (d+1)$ matrix. Then $l^\top A^{-1} l = 0$ is the equation for the set of tangent hyperplanes, where l represents the $d+1$ hyperplane coefficients. Thus, the tangent hyperplanes to a set of d balls in $\mathbb{R}^d$ can be obtained by solving the system of quadratic equations: $l^\top A_1^{-1} l = l^\top A_2^{-1} l = \cdots = l^\top A_{d+1}^{-1} l = 0$. For further details, see [38].

Let B_V be a set of $n > d+1$ imprecision regions and let $Q = \{q_1, q_2, \cdots, q_{d+1}\} \in B_V^{d+1}$. As stated in Corollary 4.5, $\mathtt{IHull}(Q)$is equivalent to $\mathtt{Conv}(Q) \setminus \mathtt{Conv}(\bigcup_{q_i \in Q^d} q_i)$. We now present a method for computing $\mathtt{IHull}(B_V)$ by first obtaining $\mathtt{IHull}(Q)$ for each $Q \in B_V^{d+1}$:

Once Algorithm1 is repeated to obtain $\mathtt{IHull}(Q)$ for each $Q \in B_V^{d+1}$, then according to Theorem 4.4, the invariant hull of B_V is computed as:

$$\mathtt{IHull}(B_V) = \mathtt{Conv}\left(\bigcup_{Q \in B_V^{d+1}} \mathtt{IHull}(Q)\right).$$

Algorithm 1 Invariant Hull Computation

$V_{IH} \leftarrow \emptyset$
$H \leftarrow \emptyset$
for each $q_i = \{q_{i_1}, q_{i_2}, \cdots, q_{i_d}\} \in Q^d$ **do**
  Compute hyperplane h_i tangent to the d members of q_i such that $q_i \subset h_i^{\text{outer}}$ and $Q \setminus f^* \subset h_i^{\text{inner}}$.
  $H \leftarrow H \cup h_i$.
end for
for each $\{h_{i_1}, h_{i_2}, \cdots, h_{i_d}\} \in H^d$ **do**
  Compute the points of intersection: $w_i = \bigcap_{n=1}^{d} h_{i_n}$.
  $V_{IH} \leftarrow V_{IH} \cup w_i$.
end for
$\texttt{IHull}(Q) \leftarrow \texttt{Conv}(\bigcup_{\forall w_i \in V_{IH}} w_i)$.

V. Centerpoint of Invariant Hulls: Safe Point Computation and Approximate Resilient Consensus

In this section, we present a method that enables normal agents to resiliently aggregate information from their neighbors by computing safe points, even in the presence of imprecise state observations and adversarial neighbors. We then apply this method to achieve approximate resilient consensus in multiagent networks. The proposed approach allows each normal agent v_i to tolerate up to $f_i(t)$ Byzantine agents in its neighborhood at time t, where $f_i(t) \leq \frac{|N_i(t)|}{d+1} - 1$. Our method builds on the Centerpoint-based method, described in Section III-A, while extending it to address state imprecision. The key innovation of our approach lies in computing safe points by *intersecting invariant hulls of imprecision regions rather than convex hulls of discrete points in* $\mathbb{R}^d$, ensuring resilience against both adversarial agents and imprecision errors. This makes our approach suitable for resilient consensus and other distributed decision-making tasks.

To clarify the fundamental principle underlying the Byzantine resilience of our method, we first examine the case without imprecision, where normal agents observe the true states of their neighbors. As discussed in [13], [30], when agents have exact state knowledge, the centerpoint of a normal agent v_i's neighbors' states serves as a safe point, provided the number of adversarial agents in the neighborhood of v_i is $f_i(t) \leq \frac{|N_i(t)|}{d+1} - 1$. This follows from a fundamental geometric property of centerpoints: specifically, the centerpoint of a set of $|N_i(t)|$ points in $\mathbb{R}^d$ lies within the convex hull of any subset containing more than $\frac{d}{d+1}|N_i(t)|$ points [39]–[41]. A key implication of this property is that if we compute the convex hull of any subset of neighbors of v_i containing more than $\frac{d}{d+1}|N_i(t)|$ points, then a centerpoint must exist within this convex hull. Given that $f_i(t) \leq \frac{|N_i(t)|}{d+1} - 1$, it follows that at least one such subset consists entirely of normal agents. Thus, a centerpoint of v_i's neighbors' states lies within the convex hull of normal agents' states, meaning that the centerpoint is a safe point.

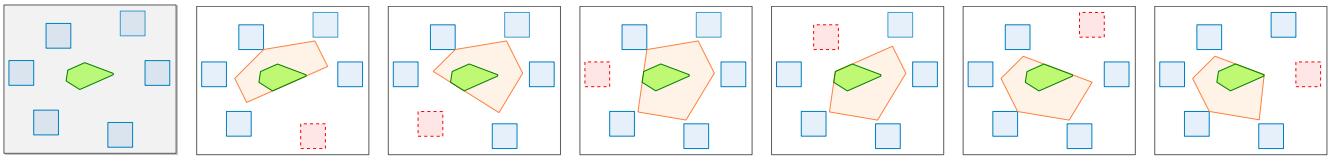
When state observations are imprecise, determining a safe point for a normal agent v_i requires a modified approach. As discussed in Section III-B, state imprecision precludes the direct use of convex hulls computed from observed states. Instead of knowing the true state of each neighbor $v_j \in N_i(t)$, agent v_i only knows that v_j's true state lies within a corresponding imprecision region. Consequently, rather than computing convex hulls of subsets of observed states, a normal agent must compute the invariant hulls of subsets of its neighbors' imprecision regions. As in the no-imprecision case, we assume that agent v_i has at most $\frac{|N_i(t)|}{d+1} - 1$ adversarial agents in its neighborhood. Since each imprecision region contains the true state of the corresponding neighbor, the invariant hull of any subset of neighbors of size $\frac{d}{d+1}|N_i(t)|$ consists of the set of points within the convex hull of every possible configuration of points within these regions, including the true state configuration. Thus, any point in the intersection of the invariant hulls of these subsets is a centerpoint of the true state configuration and, by the same reasoning as in the no-imprecision case, serves as a safe point. In other words, every point in this new centerpoint region is guaranteed to lie within the convex hull of the true states of v_i's normal neighbors. We illustrate this concept through an example below.

Example: Consider a set of $N = 6$ imprecision regions in a plane, each corresponding to an agent. Note that the true states of these agents are hidden within these imprecision regions. Among these, one imprecision region belongs to an adversarial agent, though its identity is unknown. The green-shaded region in Figure 7(a) represents the centerpoint region, obtained by intersecting the invariant hulls of subsets of imprecision regions. To see this, consider a subset of five imprecision regions, computed as $\left(\frac{d}{d+1}\right)N + 1 = 5$, and determine their invariant hull. As shown in Figures 7(b)–7(g), the green centerpoint region remains consistently enclosed within the invariant hull, regardless of which five imprecision regions are chosen. Notably, one of Figures 7(b)–7(g) must represent the actual scenario involving the single adversarial agent. In this case, the computed invariant hull, by definition, remains a subset of the convex hull formed by the true states of normal agents. Therefore, every point in the green-shaded centerpoint region, which is fully contained within the invariant hull, is guaranteed to be a safe point.

We now apply this safe point computation to the resilient vector consensus problem and introduce the *Centerpoint of Invariant Hulls (CPIH)* method, which guarantees *approximate resilient consensus* among agents in a network.

The value of $\alpha_i(t)$ is a dynamic weight that may be chosen in range $[0, 1]$ according to the application.

The CPIH algorithm identifies a region that is a generalization of the centerpoint region, in that the property of a safe point obtained through the CPIH procedure has identical properties to that of a centerpoint, with compact sets replacing the role of points. Thus each halfspace of a hyperplane that intersects a CPIH safe point contains at least $\lfloor \frac{|N_i(t)|}{d+1} \rfloor$ compact sets (imprecision regions of B_V). As such, the invariant hull of every $\frac{d|N_i(t)|}{d+1} + 1$ imprecision regions contains a safe point.



(a) (b) (c) (d) (e) (f) (g)

Fig. 7: (a) Centerpoint region based on the intersection of the invariant hulls of subsets of imprecision regions. In each of (b)–(g), the green shaded region is contained entirely in the invariant hull (orange region) of the five imprecision regions.

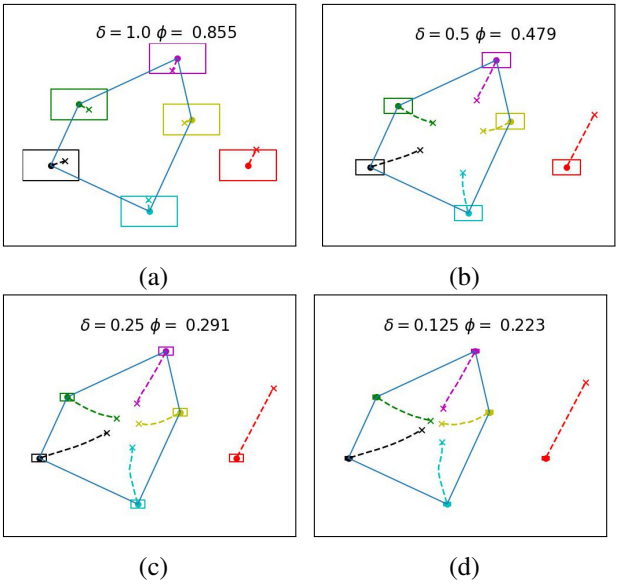
Algorithm 2 CPIH

for $t \geq 0$ **do**
  $k \leftarrow \frac{d|N_i(t)|}{d+1} + 1$
  for each $C \in B^k_{N_i}(t)$ **do**
    Compute $\texttt{IHull}(C)$. ▷ Invariant hull of each k-tuple in $B_{N_i}(t)$.
  end for
  if $\bigcap_{C \in B^k_{N_i}(t)} \texttt{IHull}(C) = \emptyset$ **then**
    Set $x_i(t+1) = x_i(t)$. ▷ Do nothing if no safe region exists.
  else
    Select a safe point $p_i(t) \in \bigcap_{C \in B^k_{N_i}(t)} \texttt{IHull}(C)$.
    Update $x_i(t+1) = \alpha_i(t)p_i(t) + (1-\alpha_i(t))x_i(t)$.
  end if
end for

A. Empirical Evaluation of CPIH

To demonstrate the CPIH algorithm's effectiveness in scenarios where the standard centerpoint-based consensus algorithm fails (see Figure 5), we conducted a series of simulations. For simplicity, we modeled imprecision regions as squares of width δ. All agents are pairwise adjacent, meaning the network graph G is complete. At each time step, agents received uniformly random state estimates from their neighbors' imprecision regions. For illustration, we selected a small range of values for δ. For each δ, normal agents followed the CPIH protocol over 5,000 time steps in the presence of a Byzantine adversary. The trajectory plots in Figure 8 depict the evolution of the true states of normal agents over time, converging to a set of points within their initial convex hull. The red trajectory corresponds to the Byzantine agent. In order to quantify the quality of the approximation we measure the ratio of the initial diameter to that of the final diameter between normal nodes. We define the diameter at time t as $diam(t) = \max_{v_i, v_j \in V_n} \|x_i(t) - x_j(t)\|$, and the denote the ratio of the final diameter at time t^* to the initial diameter at time t_0 as $\phi = \frac{diam(t^*)}{diam(t_0)}$.

As Figure 8 illustrates, the final distances between normal agents—reflecting the quality of approximate consensus—depend on the sizes of the imprecision regions. Larger imprecision regions lead to a poorer approximation, while smaller regions result in a more precise agreement. In Step 3 of the CPIH algorithm (termination condition), fixed nonzero imprecision regions prevent exact consensus, causing the algorithm to stop before agents reach a common state. In the next section, we consider a dynamic imprecision model and derive the conditions for achieving exact consensus.



(a) (b) (c) (d)

Fig. 8: Trajectory plots of approximate consensus for different imprecision region sizes: (a) $\delta = 1.0$, $\phi = 0.855$ (b) $\delta = 0.5$, $\phi = 0.479$ (c) $\delta = 0.25$, $\phi = 0.291$ (d) $\delta = 0.125$, $\phi = 0.223$ The '$\times$' marks indicate the agents' states at the algorithm's termination. The red node represents the adversarial agent, while the region enclosed by blue lines denotes the convex hull of the normal agents' initial positions.

VI. Resilient Consensus under Dynamic State Imprecision

In Section V-A, we observed that when imprecision regions are excessively large, an agent may fail to compute a safe point, even if the number of adversarial agents remains within tolerable limits (i.e., fewer than $\frac{|N_i|}{d+1}$). For the consensus problem, this limits the network to approximate consensus—where non-adversarial agents converge to a bounded region—rather than exact consensus at a single point. However, in practice, the size of these imprecision regions often depends on inter-agent distances: as agents move closer together—such as during consensus—their state observations typically become more precise, causing the imprecision regions to shrink over time. Similarly, in many localization tasks, measurement accuracy depends on both inter-agent distances and their geometric

configuration, as exemplified by the geometric dilution of precision (GDOP) effect in sensor networks [42], [43], and more broadly in triangulation-based localization systems [44]. This raises a critical question: how must these regions contract to ensure safe point computation until the network achieves its objective, such as exact consensus? In this section, we address this by modeling imprecision regions as dynamic and distance-dependent, deriving a time-varying bound that ensures convergence under our proposed method, which builds on the invariant hull framework from Section V. This will allow us to solve the following problem:

Resilient Consensus under Dynamic Imprecision – We consider a network of both normal and adversarial agents, where agent states are subject to bounded imprecision. Under the assumption that imprecision bounds are time-varying and diminish at a sufficient rate to ensure that the invariant hull of every normal agent's neighborhood is never empty, the objective of resilient consensus with imprecision is to design a distributed protocol that ensures all normal agents update their states to eventually converge on a common state within the convex hull of the initial states of normal nodes. Additionally, the protocol must satisfy the safety and agreement conditions outlined in Section II.

A. Upper Bound for the Imprecision Regions of $(d+1)$ State Vectors in $\mathbb{R}^d$

In this section, we analyze a simplified scenario to derive a sufficient condition for systematically computing a safe point under imprecision. We focus on the state vectors of $(d+1)$ agents at a given instant, which forms the basis for the more general case of $n > d+1$ agents with time-varying states in Section VI-B. This condition is determined by the relative geometric properties of the state vectors. In simpler terms, the maximum permissible magnitude of δ for $(d+1)$ agents depends on how "close" they are to collinearity–a concept we will clarify in the following discussion.

We now consider $(d+1)$ agents, whose imprecision regions are modeled as d-dimensional balls with a uniform radius δ. Our goal is to determine a sufficient upper bound on δ_i, for each v_i, such that a safe point can be computed within the convex hull of the true state vectors. Furthermore, this must be achievable for all possible state vector configurations within their respective imprecision regions. Since we focus on instantaneous state configurations in this section, we temporarily omit time-dependent notation for clarity. We begin with the following observation.

Lemma 6.1. *Let V be a set of $d+1$ agents, with corresponding set of imprecision regions B_V, such that for $v_i \in V$, B_i is a d-dimensional ball centered at x_i with radius δ. Let $\mathcal{P}(B_V)$ be defined as in Definition 4.1. If agent v_i observes configuration $R_i \in \mathcal{P}(B_V)$ and computes a safe point $\pi_i \in \mathbb{R}^d$ as a convex combination of the $d+1$ observed state values, $\pi_i = \lambda_1 r_{i(1)} + \lambda_2 r_{i(2)} + \cdots + \lambda_{d+1} r_{i(d+1)}$, where $\lambda_1 + \lambda_2 + \cdots + \lambda_{d+1} = 1$ and $\{\lambda_1, \lambda_2, \cdots, \lambda_{d+1}\}$ is a fixed set of real-valued coefficients, then the locus of all possible π_i over $R_{N_i} \in \mathcal{P}(B_V)$ is a ball of radius δ centered at $\pi^* = \lambda_1 x_1 + \lambda_2 x_2 + \cdots + \lambda_{d+1} x_{d+1}$.*

Proof. Let c_θ be an arbitrary unit vector originating from π^*. The proof follows simply by considering the interval $[a, b]$, defining the range of possible displacement along c_θ of $\|\pi^* - \pi_i\|$. Since π_i is a linear combination of the points in R_i, and the observed state $r_{i(j)} \in R_V(i)$ can be at most δ-distant from the corresponding point in X_V, it is clear that the maximum displacement occurs when all points $r_j(i) \in R_{N_i}$ are equal to $x_j + \delta c_\theta$. When this occurs, R_{N_i} is a copy of X_V translated by distance δ along c_θ, so $\|\pi^* - \pi_i\| = \delta$. Similarly the minimum distance is obtained when $r_{i(j)} = x_j - \delta c_\theta$, yielding $[a, b] = [-\delta, \delta]$. Since this is true for arbitrary c_θ, the proof is concluded. ∎

Lemma 6.1 establishes that if a point π^* is defined as a convex combination of the vertices of a simplex, and if these vertices undergo perturbations—each translated by at most a distance of δ in an arbitrary direction—then the corresponding perturbed point π (computed using the same convex combination) will deviate by at most δ from π^*. Let δ^+ denote the maximum radius of imprecision, given π^*, such that the locus of π_j, over all $r_{i(j)} \in R_i$, remains contained within $\texttt{Conv}(X_V)$, the convex hull of the true agent states. Note that any set of $d+1$ state vectors in general position in $\mathbb{R}^d$ forms a d-simplex, which we denote by S.

For an observed state x_i, let σ_i represent the d-dimensional face of S such that $x_i \notin \sigma_i$. Additionally, for any point $\pi^* \in \texttt{Conv}(X_V)$, we define $d(\pi^*, \sigma_i)$ as the minimum distance between π^* and σ_i. From Lemma 6.1, it follows that in order to guarantee that the locus of all possible π_i remains within $\texttt{Conv}(X_V)$, the imprecision bound must satisfy: $\delta^+ < \min_i d(\pi^*, \sigma_i)$. Thus, the remaining task is to determine an appropriate set of convex coefficients $\lambda_1, \lambda_2, \ldots, \lambda_{d+1}$. Every simplex has a unique point, called the *in-center* which is equidistant from all of its d-faces. The incenter's coordinates can be expressed as a convex combination, where each coefficient λ_i (the barycentric coordinate of the incenter) is weighted by the area of the $\frac{Vol(\sigma_i)}{\sum_{j \in S} Vol(\sigma_j)}$, where $Vol(\sigma)$ is the volume of the face (e.g., for a triangle with side σ_i, $Vol(\sigma_i)$ is the length). For the purpose of maximizing δ^+, in-center would give the desired values for the convex coefficients. However, we assume agents do not have access to the true state values. Consequently, they will not have access to the true in-center coefficients from one time to the next. This limitation motivates a natural choice: $\lambda_1 = \lambda_2 = \cdots \lambda_{d+1} = \lambda = \frac{1}{d+1}$. Applying these coefficients to the true positions of the simplex's vertices yields the true as the value of π^*.

Thus, given $\pi^* = \frac{1}{d+1} \sum_{i=1}^{d+1} x_i$, δ^+ must be less than the minimum distance from the centroid to any face of the simplex, which is expressed as follows:

$$\delta^+ = \min_i d(\pi^*, \sigma_i) = \min_i \frac{alt_i}{d+1},$$

where alt_i is the altitude—distance from x_i to the opposite face—(as illustrated in Figure 9). Given the choice of $\lambda = \frac{1}{d+1}$, a sufficient upper bound on δ^+ is formally stated in Proposition 6.2:

$S_i(t)$	The set of distinct $(d+1)$-tuples, containing x_i as a member, taken from the set of true agent states in the neighborhood of agent v_i at time t.
$alt_i(t)$	For a d-simplex with vertices $\{x_i, x_2, \cdots, x_{d+1}\} \subset S_i(t)$, $alt_i(t)$ is the minimum distance from vertex x_i to it's opposite face at time t. Informally referred to as "altitude".
$\theta_i(X,t)$	The minimum value of $alt_i(t)$ taken over the vertices of all simplices formed by $(d+1)$-vertex sets contained in $S_i(t)$ at time t.
$\Omega(C,t)$	For a subset of agents $C \subseteq N_i(t)$ in the neighborhood of agent v_i, $\Omega(C,t)$ is the convex hull of the centroids of observed states across all $(d+1)$-subsets of C at time t.

TABLE II: Notation for dynamic imprecision model.

Proposition 6.2. *Let V be a set of $d+1$ agents, such that each agent $v_i \in V$ receives a set of imprecise state values R_i, and $\left\| r_{i(j)} - x_j \right\| < \delta$. If each agent computes a safe point $\pi_i = \sum_{v_j \in V} \frac{r_{i(j)}}{d+1}$, then $\gamma(\pi_i) \in \mathtt{Conv}(X_V) \iff \delta < \frac{1}{d+1} \min_{v_j \in V} alt_j$, where $\gamma(\pi_i)$ denotes the locus of all possible π_i from configurations of $(d+1)$ received states within the imprecision regions.*

Note that the agent set under discussion is still of size $d+1$, and the bound on the maximum imprecision radius developed here applies only to these $d+1$ agents. However, in Section VI-B, we will demonstrate how this bound can be seamlessly incorporated into an updated version of the CPIH algorithm, which requires a set of at least $2d+2$ agents–the minimum network size needed with a single adversarial agent.

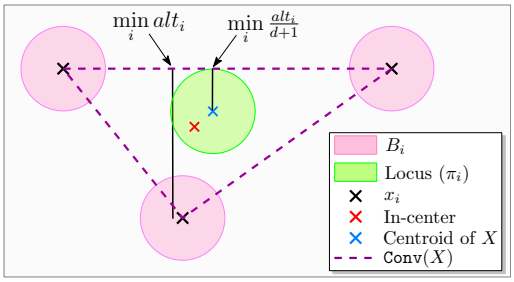


Fig. 9: Locus of π_i for $\delta^+ = \frac{1}{d+1} \min_i alt_i$ is contained within $\mathtt{Conv}(X)$.

B. Sufficient Conditions for Convergence Under DB-CPIH

We extend the sufficient upper bound from Proposition 6.2 to derive a generalized condition on imprecision radii that ensures convergence to a safe point under the *Dynamically Bounded Centerpoint Based on Invariant Hulls (DB-CPIH)* algorithm, introduced below. As a preliminary step, we explicitly state the condition under which DB-CPIH guarantees resilient consensus in the presence of imprecision. Specifically, the sufficient condition requires that, for all $v_i \in V_n$, the imprecision radius δ_i satisfies a dynamic upper bound (Definition 6.1), determined by the minimum altitude of all simplices formed from the true state values of all $(d+1)$-tuples of v_i's neighbors that include v_i.

More formally, for $v_i \in V$, let $S_i(t)$ be the set of all distinct $(d+1)$-tuples $(x_1(t), x_2(t), \cdots, x_{d+1}(t))$ such that $x_1(t) = x_i(t)$ and $x_j(t) \in S_i(t) \iff v_j \in N_i(t)$.

Definition 6.1. *($\theta-$bounded) A multiagent system is θ-*bounded *if, for every normal agent $v_i \in V_n$, the imprecision radius satisfies $\delta_i(t) < \frac{\theta_i(X,t)}{d+1}$, where*

$$\theta_i(X,t) = \min_{Q \in S_i(t)} \min_{x_k \in Q} (alt_k(t)).$$

Thus, $\theta_i(X,t)$ is the minimum altitude of all $(d+1)$-simplices of true states in the neighborhood of v_i that have $x_i(t)$ as a vertex.

If the imprecision radii of all agents satisfy this condition at all times, then, despite having only imprecise state measurements, each normal agent will always be able to identify a point within the interior of the convex hull of a set of $k = \lfloor \frac{d|N_i|}{d+1} \rfloor + 1 > d+1$ true state values. As a result, the invariant hull of any k-tuple will never be empty. Recall from Section V-A that, in the model with fixed imprecision radii, the inevitable non-existence of the invariant hull was the primary factor causing the algorithm to halt, thereby preventing exact consensus. Here, we update the imprecision model to allow for a time-varying imprecision radius, ensuring that the previously established upper bound remains satisfied at all times.

With slight modifications to the CPIH algorithm, we demonstrate that normal agents can aggregate their neighbors' states to compute safe points while achieving exact resilient consensus, even in the presence of f adversarial agents and θ-bounded imprecision. Before stating the main result, we introduce a critical definition:

Definition 6.2. *($\Omega(C,t)$): For a subset of agents $C \subseteq N_i(t)$ in the neighborhood of agent v_i, define $\Omega(C,t)$ as the convex hull of the centroids of observed states across all $d+1$-subsets of C. Formally,*

$$\Omega(C,t) = \mathtt{Conv}\left(\bigcup_{(v_{[1]}, \cdots, v_{[d+1]}) \in C^{d+1}} \frac{r_{i([1])}(t) + \cdots + r_{i([d+1])}(t)}{d+1} \right). \tag{4}$$

In (4), where $v_{[j]}$ and $r_{i([j])}$ notations are used, $[j]$ refers to the index of an arbitrary $d+1$-tuple of the neighbors of v_i, not the identity of $v_j \in V$. Moreover, for $C \subset N_i(t)$, $\Omega(C,t)$ is a point in the invariant hull of C, since $\Omega(C,t)$ satisfies the properties outlined in Theorem 4.4.

C. Dynamically Bounded Centerpoint of Invariant Hulls algorithm (DB-CPIH)

We now introduce DB-CPIH algorithm, a modification of the CPIH. The steps of the DB-CPIH are as follows:

In the DB-CPIH procedure, instead of computing invariant hulls for every set of k neighbors, agents construct convex

Algorithm 3 DB-CPIH Algorithm

for $t > 0$ **do**
  for $v_i \in V_n$ **do**
    $k \leftarrow \left\lfloor \frac{d|N_i|}{d+1} \right\rfloor + 1$
    for each $C \in N_i^k(t)$ **do**
      Compute $\Omega(C, t)$.
    end for
    Select safe point $\pi_i(t) \in \bigcap_{C \in N_i^k(t)} \Omega(C, t)$.
    $x_{v_i}(t+1) \leftarrow \alpha_i(t)\pi_i(t) + (1 - \alpha_i(t))x_{v_i}(t)$
  end for
end for

hulls using the centroids of each $(d+1)$-member subset of k observed neighbors' states. This is repeated for all k neighbors, and a point in the interior of the intersection of the resulting $\binom{|N_i(t)|}{k}$ convex hulls is selected as the safe point. When the conditions in Theorem 6.3 hold, each centroid of $d+1$ observed states lies within the invariant hull of the corresponding $d+1$ imprecision regions. It follows from Theorem 4.4 that the convex hull of the union of all $\binom{|N_i(t)|}{k}$ centroids is a subset of $\texttt{IHull}(B^k_{N_i(t)})$.In this case, $B^k_{N_i(t)}$ takes the place of $\texttt{IHull}(B_V)$ and each centroid is a point in $\texttt{IHull}(Q)$. Thus, the DB-CPIH procedure closely follows the CPIH procedure with two key differences: (a) for each $(d+1)$-member subset, the centroid is used in place of the entire invariant hull, and (b) Step 3 of the CPIH procedure in Section V is omitted, as the safe point is guaranteed to exist under the specified conditions. We now present the conditions under which the DB-CPIH algorithm guarantees exact resilient consensus.

Theorem 6.3. *Let V be a set of agents, some of which may be adversarial. For each normal agent $v_i \in V_n$ with state $x_i(t) \subset \mathbb{R}^d$ and neighborhood $N_i(t)$, where $|N_i(t)| = m_i$; the agent v_i has access to an imprecise set of state values from its neighbors, denoted as $R_i(t) = \{r_{i(1)}(t), r_{i(2)}(t), \ldots, r_{i(m_i)}(t)\}$. Normal agents updating their states according to the DB-CPIH algorithm will reach consensus if the following two conditions hold for each $v_i \in V_n$, $\forall t \geq 0$:*

1) *The set of adversarial agents in the neighborhood of v_i (i.e., $f_i(t) \subset N_i(t)$) satisfies $|f_i(t)| < \lceil \frac{|N_i(t)|}{d+1} \rceil$.*
2) *The set of normal agents, V_n, is θ-bounded.*

Proof. It is known (from [12] and [16]) that if a safe point within the interior of the convex hull of normal agents can always be selected, then agents updating their states toward this safe point at each time step will converge to a point within the convex hull of their initial positions in finite time. To complete the proof, we will show that for every $v_i \in V$, and for all $t > 0$, $\pi_i(t)$ is a safe point, where $\pi_i(t)$ is selected according to Step 3 of the DB-CPIH procedure(3). Step 1 of DB-CPIH sets the size of the subsets $C \in N_i^k(t)$ to $k = \lfloor \frac{d|N_i|}{d+1} \rfloor + 1$, which, due to the first condition of Theorem 6.3, ensures that each normal agent $v_i \in V_n$ will have a subset $C_n \subset N_i^k(t) = N_i(t)/V_f(t)$, composed exclusively of normal agents. According to Step 2 of DB-CPIH, v_i computes $\Omega(C_n, t)$ for every $C \in N_i^k(t)$. Recall that $\Omega(C_n, t)$ is the convex hull of the union of centroids of the $(d+1)$-tuples of observed states of C_n. According to Step 3 of DB-CPIH(3), π_i is chosen from the intersection $\bigcap_{C \in N_i^k(t)} \Omega(C, t)$, which implies that $\pi_i \in \Omega(C_n, t)$. It suffices to show that $\Omega(C_n, t) \subset int(\texttt{Conv}(X_n(t)))$. By Proposition 6.2, the second condition of Theorem 6.3 implies that for c, Q, where Q is a subset of $d+1$ agents in C_n, and c is the centroid of $R_Q(t)$ (the observed states of the members of Q), that $c \in int(\texttt{Conv}(X_Q(t)))$. Thus, by Theorem 4.3, $c \in int(\texttt{Conv}(X_{C_n}(t)))$. Thus, $\Omega(C_n, t)$ is the convex hull of centroids that are themselves in the interior of the convex hull of $X_{C_n}(t)$. Therefore $\Omega(C_n, t) \subset int(\texttt{Conv}(X_n(t)))$. We have shown that normal agents updating their states according to the DB-CPIH procedure, under Conditions 1 and 2 of Theorem 6.3, will always compute a safe point in the interior of the convex hull of normal neighbors, thereby ensuring that they reach consensus. This completes the proof. ∎

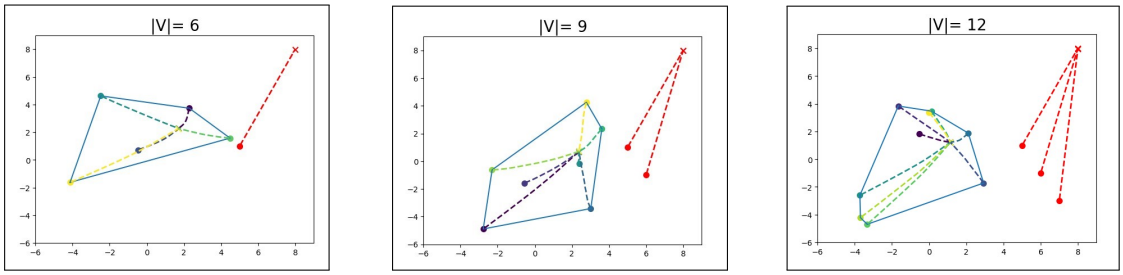
Theorem 6.3 establishes that the DB-CPIH algorithm guarantees exact consensus among normal agents when their imprecision radii satisfy the θ-bounded geometric constraint (Definition 6.1). In practical terms, this imposes a dynamic sensor precision requirement, ensuring that as imprecision shrinks over time, the states of normal agents in $\mathbb{R}^d$ converge to a single point within the convex hull of their initial positions.

Computing a safe point, even in the absence of state imprecision, remains computationally challenging due to the high-dimensional constraints of centerpoint and other geometric approaches, such as Tverberg-based methods [12], [13], [16]. In DB-CPIH, each node v_i with neighborhood size $|N_i(t)|$ computes all subsets of size $k \approx (\frac{d}{d+1})|N_i(t)|$ and calculates centroids for each $(d+1)$-tuple. While this process can be computationally expensive, it remains localized to each node's neighborhood, making it feasible in networks where agents have small neighborhoods. Moreover, fewer adversarial neighbors allow for smaller neighborhoods, further reducing computational overhead. Next, we demonstrate the performance of the DB-CPIH algorithm through simulations in $\mathbb{R}^2$.

D. Empirical Evaluation of DB-CPIH

We simulated networks of agents that simulation duration. Specifically, for each agent $v_i \in V$, the neighborhood is defined as $N_i(t) = V$ for all $t > 0$. We considered networks of sizes $|V| = 6$, 9, and 12. In each network, the number of normal agents, $|V_n|$, was determined by $|V_n| = \frac{d|V|}{d+1} + 1$, with the remaining $\frac{|V|}{d+1} - 1$ agents designated as Byzantine adversaries. Thus, for $|V| = 6$, 9, and 12, the number of adversarial agents was 1, 2, and 3, respectively, satisfying Condition 1 of Theorem 6.3. For each network configuration, we simulated the evolution of agent states over time and plotted their trajectories based on the following setup:

For each normal agent $v_i \in V_n$, the imprecision region, B_i, is a square of side length $2l$, such that $\delta_i(t) = \sqrt{2}l$ (the minimum radius of a disk containing B_i). At each time step, for every agent $v_i \in V$, $\theta_i(X, t)$ is calculated and $\delta_i(t)$ is set as $\frac{\theta_i(X,t)}{d+1}$. Normal agents calculate a safe point, $\pi_i(t)$,



(a) $|V_n| = 5$, $|V_f| = 1$ (b) $|V_n| = 7$, $|V_f| = 2$ (c) $|V_n| = 9$, $|V_f| = 3$

Fig. 10: Agent trajectories (dashed lines) as they update their states according to DB-CPIH. The adversarial agent is shown in red, while the convex hull of the normal agents' initial positions is outlined in blue. Normal agents successfully converge when imprecision decays according to the bound in Theorem 6.3.

and update their states according to the DB-CPIH algorithm. Initial positions of normal agents are randomly assigned from a uniform distribution over a unit square. The trajectories of the adversarial agents as perceived by normal agents remain entirely outside the convex hull of the normal agents' positions throughout the simulation.

Figures 10(a), (b), and (c) depict the resulting agent trajectories for each simulation, demonstrating that normal agents converge to a safe point. Figure 11(a) shows the maximum distance between the true states of any two normal agents over time, while Figure 11(b) depicts the imprecision radii (indicating the sizes of the imprecision regions) of the normal agents over time. From Figures 11(a) and (b), it is evident that the decay rate of the imprecision radii closely aligns with the rate of consensus among the normal agents.

VII. Conclusion

Many resilient distributed optimization algorithms fail when agents lack precise state values, even when the number of adversarial agents remains within tolerable limits. This failure arises from the inability to compute safe points under imprecise observations, which is critical for accurately aggregating neighbors' states in distributed tasks. In this work, we proposed a novel geometric framework that systematically models both static and dynamic imprecision, enabling the design of resilient consensus algorithms in the presence of both adversaries and noise. Building on this framework, we designed and analyzed the DB-CPIH algorithm, which provides resilience guarantees even under uncertainty, and rigorously characterized conditions for consensus among normal agents.

In future work, we aim to analyze the performance of resilient consensus algorithms when resilience conditions are *intermittently* violated. The DB-CPIH algorithm currently guarantees convergence of normal nodes when the number of adversarial neighbors satisfies $|f_i(t)| < \lceil \frac{N_i(t)}{d+1} \rceil$ for each $v_i \in V_n$ at all times, and when the imprecision regions remain θ-bounded throughout the execution. However, in practical settings, these conditions may be temporarily violated. For instance, the number of adversarial neighbors or the size of imprecision regions may exceed their respective thresholds over finite time intervals. Preliminary observations suggest that normal agents may still reach consensus during such transient

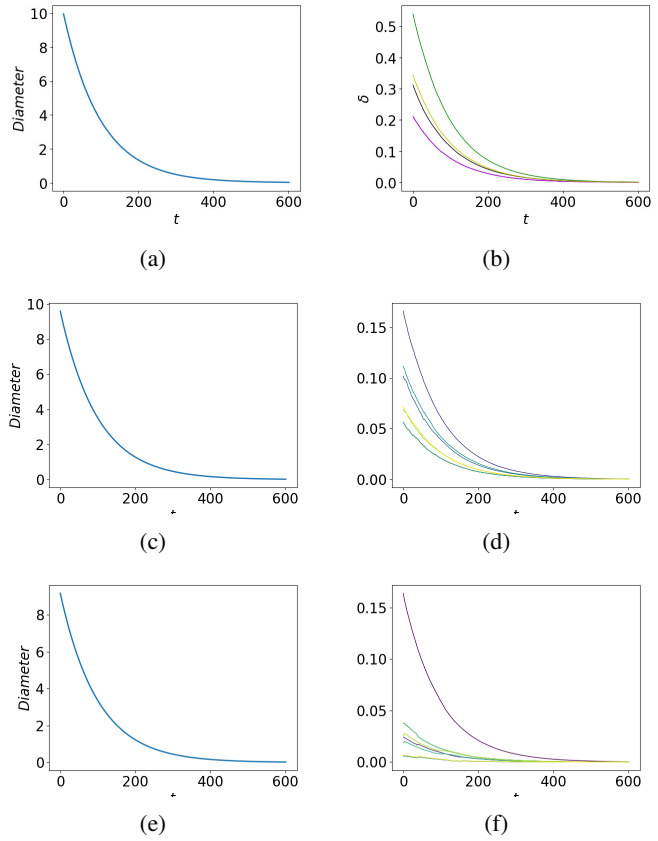


Fig. 11: Plots in (a), (c), and (e) show the maximum distance between any two normal agents over time, given by $\max_{v_i, v_j \in V_n} \|x_i(t) - x_j(t)\|$ (i.e., diameter), for network sizes $|V| = 6$, 9 and 12, respectively. Plots in (b), (d), and (f) illustrate the imprecision radii for each agent over time for the same network sizes.

violations without compromising safety. However, the extent and duration of allowable violations that preserve convergence guarantees remain unknown. Characterizing the precise conditions under which these constraints can be relaxed while still ensuring safety and eventual agreement is a promising direction for future investigations.

REFERENCES

[1] C. A. Lee and W. Abbas, "A geometric approach to resilient distributed consensus accounting for state imprecision and adversarial agents," in *American Control Conference (ACC)*, 2024, pp. 220–225.

[2] H. Ishii, Y. Wang, and S. Feng, "An overview on multi-agent consensus under adversarial attacks," *Annual Reviews in Control*, vol. 53, pp. 252–272, 2022.

[3] M. Pirani, A. Mitra, and S. Sundaram, "Graph-theoretic approaches for analyzing the resilience of distributed control systems: A tutorial and survey," *Automatica*, vol. 157, p. 111264, 2023.

[4] A. Mitra and S. Sundaram, "Byzantine-resilient distributed observers for lti systems," *Automatica*, vol. 108, p. 108487, 2019.

[5] T. Yu, R. C. de Lamare, and Y. Yu, "Robust resilient diffusion over multi-task networks against byzantine attacks: Design, analysis and applications," *IEEE Transactions on Signal Processing*, vol. 70, pp. 2826–2841, 2022.

[6] Y. Chen, S. Kar, and J. M. Moura, "Resilient distributed parameter estimation with heterogeneous data," *IEEE Transactions on Signal Processing*, vol. 67, no. 19, pp. 4918–4933, 2019.

[7] J. Li, W. Wang, W. Abbas, and X. Koutsoukos, "Distributed clustering for cooperative multi-task learning networks," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 6, pp. 3933–3942, 2023.

[8] Z. Yang, A. Gang, and W. U. Bajwa, "Adversary-resilient distributed and decentralized statistical inference and machine learning: An overview of recent advances under the byzantine threat model," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 146–159, 2020.

[9] S. Yu and S. Kar, "Secure distributed optimization under gradient attacks," *IEEE Transactions on Signal Processing*, vol. 71, pp. 1802–1816, 2023.

[10] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4289–4305, 2012.

[11] K. Kuwaranancharoen and S. Sundaram, "On the geometric convergence of byzantine-resilient distributed optimization algorithms," *SIAM Journal on Optimization*, vol. 35, no. 1, pp. 210–239, 2025.

[12] H. Park and S. A. Hutchinson, "Fault-tolerant rendezvous of multirobot systems," *IEEE Transactions on Robotics*, vol. 33, pp. 565–582, 2017.

[13] W. Abbas, M. Shabbir, J. Li, and X. Koutsoukos, "Resilient distributed vector consensus using centerpoint," *Automatica*, vol. 136, 2022.

[14] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 766–781, 2013.

[15] L. Su and N. Vaidya, "Multi-agent optimization in the presence of byzantine adversaries: Fundamental limits," in *American Control Conference (ACC)*, 2016, pp. 7183–7188.

[16] H. Mendes, M. Herlihy, N. Vaidya, and V. K. Garg, "Multidimensional agreement in byzantine systems," *Distributed Computing*, vol. 28, no. 6, pp. 423–441, 2015.

[17] J. Yan, X. Li, Y. Mo, and C. Wen, "Resilient multi-dimensional consensus in adversarial environment," *Automatica*, vol. 145, 2022.

[18] J. Yan, Y. Mo, X. Li, and C. Wen, "A "safe kernel" approach for resilient multi-dimensional consensus," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 2507–2512, 2020.

[19] Y. Wang, H. Ishii, F. Bonnet, and X. Défago, "Resilient consensus for multi-agent systems under adversarial spreading processes," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 5, pp. 3316–3331, 2022.

[20] J. Li, W. Abbas, and X. Koutsoukos, "Resilient distributed diffusion in networks with adversaries," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 6, pp. 1–17, 2019.

[21] L. Su and N. H. Vaidya, "Byzantine-resilient multiagent optimization," *IEEE Transactions on Automatic Control*, vol. 66, pp. 2227–2233, 2020.

[22] J. Zhu, Y. Lin, A. Velasquez, and J. Liu, "Resilient distributed optimization," in *American Control Conference (ACC)*, 2023, pp. 1307–1312.

[23] K. Kuwaranancharoen, L. Xin, and S. Sundaram, "Byzantine-resilient distributed optimization of multi-dimensional functions," in *American Control Conference (ACC)*, 2020, pp. 4399–4404.

[24] J. Usevitch and D. Panagou, "Resilient leader-follower consensus to arbitrary reference values in time-varying graphs," *IEEE Transactions on Automatic Control*, vol. 65, no. 4, pp. 1755–1762, 2019.

[25] M. Safi, S. M. Dibaji, and M. Pirani, "Resilient coordinated movement of connected autonomous vehicles," *European Journal of Control*, vol. 64, p. 100613, 2022.

[26] S. M. Dibaji and H. Ishii, "Resilient consensus of second-order agent networks: Asynchronous update rules with delays," *Automatica*, vol. 81, pp. 123–132, 2017.

[27] Z. Liao, J. Shi, S. Wang, Y. Zhang, R. Chen, and Z. Sun, "Dynamic event-triggering resilient coordination for time-varying heterogeneous networks," *IEEE Transactions on Signal and Information Processing over Networks*, pp. 1–13, 2025.

[28] Z. Yang and W. U. Bajwa, "Byrdie: Byzantine-resilient distributed coordinate descent for decentralized learning," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 4, pp. 611–627, 2019.

[29] L. Yuan and H. Ishii, "Asynchronous approximate byzantine consensus: A multi-hop relay method and tight graph conditions," *Automatica*, vol. 171, p. 111908, 2025.

[30] J. Li, W. Abbas, M. Shabbir, and X. Koutsoukos, "Byzantine resilient distributed learning in multirobot systems," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3550–3563, 2022.

[31] D. Jayasimha, "Fault tolerance in a multisensor environment," in *Proceedings of IEEE Symposium on Reliable Distributed Systems*, 1994, pp. 2–11.

[32] E. F. Nakamura, A. A. Loureiro, and A. C. Frery, "Information fusion for wireless sensor networks: Methods, models, and classifications," *ACM Computing Surveys*, vol. 39, no. 3, pp. 9–es, 2007.

[33] M. Löffler, *Data imprecision in computational geometry*. Utrecht Univesity, 2009.

[34] J. Park, R. Ivanov, J. Weimer, M. Pajic, S. H. Son, and I. Lee, "Security of cyber-physical systems in the presence of transient sensor faults," *ACM Transactions on Cyber-Physical Systems*, vol. 1, pp. 1–23, 2017.

[35] G. Frehse, A. Hamann, S. Quinton, and M. Woehrle, "Formal analysis of timing effects on closed-loop properties of control software," in *IEEE Real-Time Systems Symposium*, 2014, pp. 53–62.

[36] I. Bárány, "A generalization of carathéodory's theorem," *Discrete Mathematics*, vol. 40, no. 2, pp. 141–152, 1982.

[37] M. Abrahamsen and B. Walczak, "Common tangents of two disjoint polygons in linear time and constant workspace," *ACM Trans. Algorithms*, vol. 15, no. 1, dec 2018.

[38] J. Richter-Gebert, *Perspectives on Projective Geometry: A Guided Tour Through Real and Complex Geometry*. Berlin, Heidelberg, New York: Springer, 2011.

[39] J. Pach and P. K. Agarwal, *Combinatorial Geometry*. John Wiley & Sons, 2011.

[40] J. Matousek, *Lectures on Discrete Geometry*. Springer Science & Business Media, 2013.

[41] S. Ray and N. Mustafa, "An optimal generalization of the centerpoint theorem, and its extensions," in *Annual Symposium on Computational Geometry*, 2007, pp. 138–141.

[42] J. Zhang and J. Lu, "Analytical evaluation of geometric dilution of precision for three-dimensional angle-of-arrival target localization in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 16, no. 5, pp. 1–14, 2020.

[43] S. Kumar and R. M. Hegde, "Indoor node localization using geometric dilution of precision in ad-hoc sensor networks," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, 2014, pp. 1525–1529.

[44] O. Tekdas and V. Isler, "Sensor placement for triangulation-based localization," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 681–685, 2010.