Community Consensus: Converging Locally despite Adversaries and Heterogeneous Connectivity

Cristina Gava, Áron Vékássy, Matthew Cavorsi, Stephanie Gil, Frederik Mallmann-Trenn

***Abstract*— We introduce the concept of *community consensus* in the presence of malicious agents using a well-known median-based consensus algorithm. We consider networks that have multiple well-connected regions that we term *communities*, characterized by specific robustness and minimum degree properties. Prior work derives conditions on properties that are necessary and sufficient for achieving *global* consensus in a network. This, however, requires the minimum degree of the network graph to be proportional to the number of malicious agents in the network, which is not very practical in large networks. In this work, we present a natural generalization of this previous result. We characterize cases where, although global consensus is not reached, some subsets of agents V_i will still converge to the same values $\mathcal{M}_i$ among themselves. To reach this new type of consensus, we define more relaxed requirements in terms of the number of malicious agents in each community, and the number k of edges connecting an agent in a community to agents external to the community.**

Keywords: Consensus, Distributed Model, Malicious Agents, MCA, Communities

I. Introduction

In this work we present a new form of local consensus in a network, that we call *community consensus*, that can be achieved even in the presence of malicious agents. We focus on the case where legitimate agents in different "communities" of a network try to reach consensus within their respective subgraph. We characterize conditions on the graph topology such that community consensus can be reached despite the presence of malicious agents. Importantly, community consensus can still be attained in many networks where global consensus [1] is not possible.

The need for multiple agents to agree on a value in a distributed manner is crucial for many real-life applications, from systems of multiple sensors to social dynamics. The topology and connectivity of the network, though, can impede that agents receive information from any region of the network, preventing common agreement. Conversely, there can be scenarios where the ultimate goal is not to have an entire network to agree on a unique value, but instead having different regions of the network to agree on different values – which is the focal point of this paper. We consider a network where agents communicate in a distributed manner and they can query the values of all their neighbors. However, they do not know whether other agents are malicious nor the structure of the network outside of their neighborhood.

Many works in the literature focus on consensus problems in networks under a graph-theoretic and stochastic perspective, from average consensus [2], [3] to opinion dynamics [4], [5]. Even malicious agents are considered in some works ([6]–[9]), but little attention has been posed on networks having heterogeneous connectivity and malicious agents. Some of these works also focus on devising strategies to identify malicious agents or exclude them from the process of reaching consensus [10], [11]. Here, identifying malicious agents is not a primary focus, instead, we focus on characterizing conditions to reach community consensus with unknown malicious agents in the network.

We build upon the model devised in [1], which aims for all legitimate agents in the network to converge to the same value. In [1], the authors require a strong notion of robustness that may not be applicable to many practical use cases, where these stringent connectivity constraints are not fulfilled. We use the same median consensus algorithm, but pose a different goal, in that we aim for legitimate agents within a subgraph of the network to converge to a value that is within the convex hull of their initial values, i.e., *community consensus*. Relaxing the goal also allows us to relax the connectivity constraints. However, reaching community consensus is non-trivial for two important reasons: 1) consensus still requires sufficient robustness within a community, but also 2) community consensus requires some degree of isolation between communities. Specifically, a characterization of connectivity within the community and across communities that depends on the number of malicious agents in each community is required.

Towards the goal of characterizing conditions when community consensus can be achieved, our work addresses several challenges: for some community i, legitimate agents in a subset V_i of the network need to be able to reach the same fixed point $\mathcal{M}_{V_i} \in \mathbb{R}$, and they need to be robust outliers, with respect to the values in V_i. For the edge case of the entire network being a single community, [1] already provides necessary and sufficient conditions for this to happen. However, with multiple communities connected to each other, our analysis has to take into account the effect these communities can have on each other. In this work, we characterize the parameters of a network G such that community consensus is attainable. As we will show in Section VI, this is done in terms of the number of edges k_i connecting agents in subset V_i to other subsets, and in terms

Cristina Gava and Frederik Mallmann-Trenn are with the Department of Informatics, King's College London, WC2R 2LS London, U.K. (e-mail:frederik.mallmann-trenn@kcl.ac.uk; cristina.gava@kcl.ac.uk). Áron Vékássy, Matthew Cavorsi and Stephanie Gil are with the School of Engineering, Applied Sciences at Harvard University, Allston, MA 02134 USA (e-mail: avekassy@g.harvard.edu; mcavorsi@g.harvard.edu; sgil@seas.harvard.edu).

The authors gratefully acknowledge AFOSR grant #FA9550-22-1-0223, NSF grant #CNS-2147694 and EP/W005573/1 for partial support of this work.

of the number of malicious nodes f_i in V_i. The strength of our approach is that it models scenarios in which the network connectivity is not homogeneous.

This paper is structured as follows. In Section II, we provide the current state of the art and explain how our work places itself in it, also recalling, in Section III, important notions of network robustness that we make use of. Our framework is described in Section IV and in Section V, while in Section VI we present our theoretical results and we prove them. Empirical results are presented in Section VII, precisely showing the necessity for the constraints in our main theorem. We finish this work with describing some relevant use cases in Section VIII and drawing final conclusions in Section IX.

II. Related Literature

Consensus problems on evolving networks have been thoroughly studied in many fields, from opinion dynamics (See for example [4], [5], [12] and the references therein) to first order distributed optimization algorithms [13], [14], where linear matrix equations and Lyapunov functions are used. These works present the so-called *average consensus* protocols, where agents update their value by averaging it with the value of their neighbors. Works on average consensus have thoroughly developed under many points of view: From fixed and switching topologies, [15]–[17], to a graphical approach [3], and to a Markov Chain approach [2], [18], where worst-case convergence rates and graph-theoretic conditions have been examined. Aside from the average, other aggregates prove useful and more robust to outliers. Fundamental is the seminal work from Kempe [19], which, with probability $1-\delta$, shows an ε-convergence in the *push-sum* gossip-based protocol in $O(\log n)$. In it, authors study the problems of computing several types of aggregates through gossip-based protocols. A follow up work is [20], where the authors focus on an asynchronous way to find the k^{th} smallest value in a network of n agents. In this work, authors concentrate on the median, which is a valuable aggregate, because of its robustness to outliers. In the control theory community, some works explored different robust aggregates besides median and mean: some methods extend to tools such as Krum and multi-Krum [21], geometric median [22], coordinate-wise median, coordinate-wise trimmed mean [23], Bulyan and multi-Bulyan [24]. A fundamental aspect of our work is the presence of adversarial agents in the network. While the presence of wrong values in a network has been previously considered, the literature around consensus models with adversaries (i.e., the number of malicious or spoofed agents) is still relatively understudied. Many recent works on adversarial agents pose the attention on leader-follower dynamics [9] where the value of one agent is taken as reference for the other agents to follow. Our work applies to a leaderless dynamic, where agents possibly reach a common value but are unaware of the global status of the network (decentralized approach) and of the fixed point. The existing results on resilient cooperative control are, however, still conditioned on many communication network's connectivity requirements. In the case of [25], the authors show robustness of the network of size n to $o(\sqrt{n})$ byzantine agents, while [26] looks at the case up to $\sqrt{n}$ agents can be corrupted in the network at any time. In [27] authors design what they call a "Stabilizing Consensus" able to tolerate $f < n$ crash faults and $f\ s.t.\ 3f < n$ byzantine faults in a network of n agents. These works leverage on asynchronous models, and there is no focus on the underlying network topology, but instead, interaction among agents is modeled as random matchings. Our work compares to the work of Zhang and Sundaram [1] and builds from that. The authors look at a synchronous, distributed, median consensus model where f malicious agents can be present in a network of n agents. To show the validity of the model, authors devise a new notion of robustness, namely *(r,s)-excess robustness* and pair it with the requirement that the minimum degree of the network be $d_{min} \geq 2f + 1$. Their work is a further development from previous work in [28] and [29].

III. Background: Robustness

A. The Median Consensus Algorithm (MCA)

The authors of [1] present a consensus protocol where agents in a network follow a synchronous, distributed consensus algorithm called MCA. We refer to it as *median-based consensus protocol*. In the MCA, every legitimate agent u in a graph G holds a value. At every step t, this value is averaged with the median of the values from u's neighbors. The algorithm iterates perpetually and the authors prove that the agents will eventually reach a type of consensus they call *Resilient Asymptotic Consensus (RAC)* (defined in Definition 5). Due to the presence of malicious agents, legitimate agents can only reach consensus under certain assumptions of robustness of the graph. To this end, [1] introduces (r, s)-*excess robustness*, building from the notion of r-*excess robustness* in the previous works. The notion of r-excess robustness conveys the idea that, for any partition of the network, at least one agent in the partition has r more neighbors outside the partition than inside; (r, s)-excess robustness asks that there always be at least s of such agents. We recall here the notions of robustness presented in [1]. In our approach, we use the same notions and observe their application to sub-graphs induced in the original graph.[1]

B. r-excess robustness and (r, s)-excess robustness

The median is a robust statistic that is not influenced by outliers. However, this also means that legitimate values at the extremes of an ordered vector of values may never be selected as the median. Consider the example of a graph G where agents can be partitioned into two sets $\mathcal{A}$ and $\mathcal{B}$, such that, at time $t = 0$, any agent $u \in \mathcal{A}$ holds a value a and any agent $v \in \mathcal{B}$ holds $b \neq a$. Assuming agents share and update their values following the MCA, they will not reach convergence if every agent has more neighbors in its own subset than neighbors in the other subset – no agent will change its opinion, regardless of the

[1]We recall here that an *induced sub-graph* on $G = (V, E)$ is a graph $G[V^*] = (V^*, E^*)$ where $V^* \subseteq V$ and E^* contains *all* of the edges, from the original graph G, that connect the agents in V^*.

connectivity in other parts of the graph. Stronger assumptions on connectivity are hence needed, as provided by the notion of r-excess robustness, from [28] and [1]. Let $\mathcal{N}_u$ denote the neighborhood of agent u.

Definition 1 (r-excess reachable set) *Given a graph $G = (V, E)$ and a nonempty subset S of agents of G, we say S is an r-excess reachable set if $\exists u \in S$ such that $|\mathcal{N}_u \setminus S| - |\mathcal{N}_u \cap S| \geq r$, where $r \in \mathbb{Z}_{\geq 0}$. When clear from the context, we will also say that agent u (with regard to set S) is r-excess reachable.*

Definition 2 (r-excess robust graph) *A graph $G = (V, E)$ is r-excess robust, with $r \in \mathbb{Z}_{\geq 0}$, if for every pair of nonempty, disjoint subsets of V, at least one of the subsets is r-excess reachable.*

However, it might happen that all the reachable agents are also malicious, resulting in the network being disconnected and preventing its legitimate agents from reaching consensus. To address this case, in [1], robustness is expanded to (r, s)-excess robustness, with the idea to ensure that there always are more reachable agents than malicious agents.

Definition 3 ((r, s)-excess robustness) *Take $r \in \mathbb{Z}_{\geq 0}$ and $s \in \{1, ..., n\}$. A graph $G = (V, E)$ is (r, s)-excess robust if*

- *For every pair of nonempty, disjoint subsets $S_1, S_2 \subseteq V$*
- *Given the set $\mathcal{X}^r_{S_i} = \{u \in S_i : |\mathcal{N}_u \setminus S_i| - |\mathcal{N}_u \cap S_i| \geq r\}$ for $i \in \{1, 2\}$*

At least one of the following holds: $|\mathcal{X}^r_{S_1}| + |\mathcal{X}^r_{S_2}| \geq s$; $|\mathcal{X}^r_{S_1}| = |S_1|$; $|\mathcal{X}^r_{S_2}| = |S_2|$.

IV. Problem Formulation

The previous definitions leverage on the property that all the agents in the graph need to be somewhat *connected enough* in order for the MCA to work. In other words, this means assuring that any subset of agents $S \subseteq V$, as defined in Definition 1, is able to incorporate values from agents in $V \setminus S$ in the update step. In this section, we apply the notions of consensus and robustness from [1] to more general networks, where robustness properties pertain to induced sub-graphs of the starting graph G, rather than its entirety. We formalize the case of a graph with heterogeneous connectivity, where more connected agents belong to subsets that fulfill (r, s)-excess robustness criteria. In these subsets, legitimate agents can reach consensus and not only are they robust to malicious agents in the same subset, but to malicious agents in the whole G as well.

A. Graph Structure and Additional Notation

We consider a graph $G = (V, E)$. Let $n = |V|$. In G, we denote by *legitimate* the agents that follow the MCA and by *malicious* all the others. We call L the set of legitimate agents in G and F the set of malicious agents, such that $F \cup L = V$ and $F \cap L = \emptyset$. We define a partition of V: $\{V_1, V_2, \ldots, V_c\}$ into c subsets, where V_i indicates the generic i^{th} subset. For any subset $V_i \subseteq V$ and an agent $u \in V_i$, we call $\mathcal{N}_u = \{v \mid (u, v) \in E\}$ and $k_i = \max_u\{|\mathcal{N}_u \setminus V_i|\} \geq 0$. We use the notation $G[V_i]$ to indicate the sub-graph of G induced by the subset of agents in V_i. Further, an agent u has degree d_u, G's minimum degree is denoted by $d_{min} = \min_u\{d_u\}$, and the minimum degree of $G[V_i]$ is denoted by $d^{V_i}_{min}$.

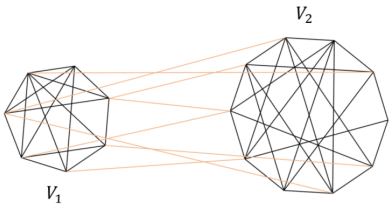


Fig. 1: Example of graph where two subsets of agents V_1 and V_2 are visibly more connected within themselves. Lower connectivity between V_1 and V_2 is represented by the orange edges.

The process unfolds in discrete rounds, starting from $t = 0$. We represent with $\xi_u(t) \in \mathbb{R}$ the value of an agent u at time t, let it be legitimate or malicious. We assume that legitimate agents do not know which agents are malicious, nor which values they can take. However, malicious agents may know each other and, therefore, distinguish between legitimate and other malicious agents.

We do not make assumptions on the malicious values, allowing us to account for the fact that malicious agents can be able to coordinate their values in such way that their action is the most disruptive. This includes being able to potentially update their value in later stages. Recall that $\mathcal{N}_u$ indicates the neighborhood of u, we call $\boldsymbol{\xi}^{(u)}(t) = \{\xi_v(t) : v \in \mathcal{N}_u\}$ the *ordered* vector of values held by the neighbors of u at time t. When there is no ambiguity, we will omit the time reference t. Finally, using the notation $\boldsymbol{\xi}^{(u)}_{[i]}$ to indicate the i^{th} entry of the vector $\boldsymbol{\xi}^{(u)}$, we consider the median operator

$$M\left(\boldsymbol{\xi}^{(u)}\right) = \begin{cases} \frac{\boldsymbol{\xi}^{(u)}_{[d_u/2]} + \boldsymbol{\xi}^{(u)}_{[d_u/2+1]}}{2} & \text{for } d_u \text{ even} \\ \boldsymbol{\xi}^{(u)}_{[(d_u+1)/2]} & \text{otherwise.} \end{cases}$$

B. Fault Model

We recall that F is the set of malicious agents. Similarly to V, we define a partition of $F \subset V$:

$$F = \{F_1, F_2, \ldots, F_c | F_i \subset V_i \text{ and } |F_i| = f_i, \forall i \in [1, c]\},$$

Where we use f_i to denote the number of malicious agents in the specific subset V_i. From the literature, we present three types of fault models: *F-total*, *F-local* and *Byzantine* models. The first two models require that at most $|F|$ malicious agents be present in the whole network (F-total) or in an agent's neighborhood (F-local) at any point in time. In the latter case, agents are still organized under an F-total or F-local scheme, though they are Byzantine agents. Our malicious agents are Byzantine, in that they can coordinate to strategically send a specific set of values to their neighbors. Our model is *F-total* with respect to the whole agent set V.

C. Community Structure

As mentioned above, we look at a graph G within which there can be multiple regions satisfying specific connectivity constraints. We call these regions *communities* and formally introduce them in Definition 4.

Definition 4 ((k_i, f_i)-community) *Consider a graph $G = (V, E)$ and a subset of agents $V_i \subseteq V$. For $k_i, f_i \geq 0$, the induced sub-graph $G[V_i]$ is a* (k_i, f_i)-community *if:*

- *$G[V_i]$ is $(k_i, f_i + 1)$-excess robust;*
- *Its minimum degree is $d_{min}^{V_i} \geq 2f_i + k_i + 1$*

Definition 4 characterises a sub-graph of G that is (r, s)-excess robust, however it does not imply anything about the robustness of G. Note that, if G is (r, s)-excess robust, then $V_i = V$, as well as the fact that communities can contain smaller communities within themselves. Furthermore, if $G[V_i]$ is a (k_i, f_i)-community, for $V_i' \subset V_i$, not necessarily is $G[V_i']$ a (k_i, f_i)-community. For example, take $f_i' = 0, r > 0$. There might exist a subset $V_i' = \{u, v\}$ such that $(u, v) \notin E$: in this case, $G[V_i']$ is clearly not $(r, 1)$-excess robust, however, u and v may be connected to other agents in V_i so that robustness of $G[V_i]$ may be guaranteed all the same. Conversely, for $V_1, V_2 \subset V$ such that $V_1 \cap V_2 = \emptyset$ and for $G[V_1]$ and $G[V_2]$ two communities, $G[V_1 \cup V_2]$ is not necessarily a community. A trivial example is two $(k_i, f_i + 1)$-excess robust graphs being disconnected from each other.

V. The Median Consensus Algorithm and RAC

We apply the same algorithm presented in [1] to the graph G as described in Section IV-A. Every agent $u \in V$ starts with a value $\xi_u(0) \in \mathbb{R}$, such that, at any subsequent timestep t, the value $\xi_u(t)$ follows the same update step presented in the MCA introduced in [1]. Namely, for $\alpha \in (0, 1)$

$$\xi_u(t+1) = \alpha \xi_u(t) + (1-\alpha) M(\boldsymbol{\xi}^{(u)}(t)) \quad \text{if } u \in L \tag{1}$$

This update step dictates that all the legitimate agents in the network will update their value to an average between their current value and the median of their neighbourhood. This is done synchronously for every agent $u \in L$. No assumptions are made on the update of the values for the malicious agents, nor on the value that they communicate to legitimate agents.

We define ξ_{m_i} and ξ_{M_i} as, respectively, $\min\{\xi_u(0) \mid u \in V_i\}$ and $\max\{\xi_u(0) \mid u \in V_i\}$. We therefore recall the definition of *RAC* from [1] and expand it to account for this new notation.

Definition 5 (Resilient Asymptotic Consensus) *Under any of the fault models and for a subset $V_i \in V$, the legitimate agents in V_i are said to achieve* RAC *if* both *of the following conditions are satisfied for* any *choice of initial values $\xi_u(0) \in \mathbb{R}$.*

- Agreement Condition*: there exists $\mathcal{M}_i \in \mathbb{R}$ such that $\lim_{t\to\infty} \xi_u(t) = \mathcal{M}_i$;*
- Safety Condition*: The values of agents are throughout between the minimum and maximum values of the legitimate agents, i.e., for $t \in \mathbb{Z}_{\geq 0}$, $\xi_u(t) \in [\xi_{m_i}, \xi_{M_i}]$*

VI. Theoretical Results

The framework we introduced in the previous sections allows us to now present the results of our work . We start by stating the core result in Theorem 1, followed by an insight on the structure of its proof. Proposition 1 follows, providing us with all the elements needed to later prove Theorem 1. For $u \in V_i$, recall that $k_i = \max_u\{|\mathcal{N}_u \setminus V_i|\} \geq 0$ is the number of edges going from the node u to a set outside of V_i. We call these "external edges."

Theorem 1 *Take $G = (V, E)$, and the partition of V in c subsets $V_1, V_2, \ldots, V_c$. For a given V_i with f_i malicious agents, and where each $u \in V_i$ has at most $k_i \geq 0$ external edges, if $G[V_i]$ is a (k_i, f_i)-community, i.e.,*

1) *$G[V_i]$ is $(k_i, f_i + 1)$-excess robust*
2) *$G[V_i]$ has $d_{min}^{V_i} \geq 2f_i + k_i + 1$*

holds, and

3) *All legitimate agents in $G[V_i]$ run MCA*

Then, every legitimate agent in V_i will reach Resilient Asymptotic Consensus (cf. Definition 5).

Remark 1 *Agents of different communities may converge to different values.*

Remark 2 *If $k_i = 0$ and $c = 1$, which means that the graph consists of one single big community, this theorem is equivalent to Theorem 1 in [1].*

Theorem 1 shows that the use of MCA allows for the relaxation of two core prerequisites needed when applying it to reach community consensus. First, G does not need to be (r, s)-excess robust in order to have subsets of agents reach consensus within $G[V_i]$ through MCA; instead, it needs to have subsets of agents to be *connected enough* by their induced sub-graphs being (k_i, f_i+1)-excess robust. Second, the minimum degree requirement for a node $u \in V_i$ is only dependent on the number of malicious agents that are in V_i and the number of neighbors u has outside of its community. This is much less restrictive than the degree requirement for global consensus, which is dependent on the total number of malicious agents in the entire network. This implies that communities with fewer agents can also have a lower minimum degree. Simulations in Section VII further confirm the theoretical results and the necessity of our assumptions.

A. Proof Idea

The full proof of Theorem 1 can be found in Section VI-C and is structured as follows. We consider a graph G, as formalized in Section IV-A, and an induced sub-graph $G[V_i]$. We show the following (Proposition 1). Consider V_i: The agents within V_i that are highly connected to other agents within the same subset still have more connections to agents within V_i compared to outside V_i if we look at G instead of $G[V_i]$. More formally, if $G[V_i]$ is a (k_i, f_i)-community, then we show that agents that are k_i-excess reachable in $G[V_i]$ maintain high-enough reachability in G – even when connected to k_i new neighbors outside V_i. Therefore, we can apply the results from [1] (cf. Theorem 1 [1]) to show that agents in the community $G[V_i]$ will reach resilient asymptotic consensus.

B. Formal Analysis

We start by defining the *isolation* property of a community: it characterizes the case where agents in V_i can be con-

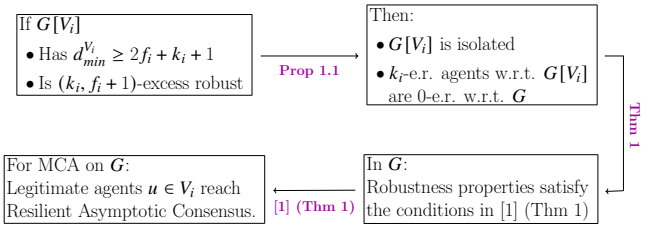


Fig. 2: Flow of the proofs.

nected to agents outside of V_i, i.e., $v, w, \ldots \in V \setminus V_i$, without resulting in the induced sub-graph being a community (cf. Definition 4).

Definition 6 (Community Isolation) *Consider a community $G[V_i]$, an agent $u \in V_i$ and a neighboring agent $v \in \mathcal{N}_u \setminus V_i$. $G[V_i]$ is* isolated *if, in the update step in Eq.* (1) *for u, any $\xi_v(t)$ that does not satisfy the* safety condition *does not influence the median operator.*

Making use of this, the following proposition characterizes how the excess reachability of agents within a community compares to the excess reachability within the graph.

Proposition 1 *Take $G = (V, E)$ and a partition of V into c subsets $\{V_1, V_2, \ldots, V_c\}$. Let the induced sub-graph $G[V_i]$ be a (k_i, f_i)-community. Let u be a k_i-excess reachable agent in $G[V_i]$ w.r.t. a set $S \subset V_i$. Then, u is 0-excess reachable in G w.r.t. all sets $S \cup N$, where $N \subseteq \mathcal{N}_u \setminus V_i$. Furthermore, $G[V_i]$ is isolated.*

Before we prove this formally, it is worth emphasizing that the statement is non-trivial: First, agents in V_i do not necessarily maintain the same reachability when looked at w.r.t. the whole graph G. Even in the absence of malicious agents, new edges connecting to external values could influence an agent $u \in V_i$ by making it diverge from the $\mathcal{M}_i$. Second, having more edges potentially means more malicious agents influencing a legitimate agent. It is therefore paramount that the cross-community edges do not expose any agent in a community to too many malicious agents.

Proof: [Proof of Proposition 1] In the first part we prove that the agents that are 0-excess reachable w.r.t. $G[V_i]$ are 0-excess reachable w.r.t. G. In the second part, we show that $G[V_i]$ is isolated.

a) Robustness Property

Without loss of generality, we exclude the trivial case for which $V_i = V$ and $k_i = 0$, and consider the community $G[V_i]$, $V_i \subset V$. We take an agent $u \in V_i$ and an agent $v \in V \setminus V_i$, and observe the connectivity of the induced sub-graph $G[V_i^*] = G[V_i \cup \{v\}]$. By definition, any r-excess reachable agent is also 0-excess reachable. Key in this part is to show that any change in the connectivity of $G[V_i^*]$ will not lead to the number of 0-excess reachable agents in V_i to decrease. We therefore take a subset of agents $S \subseteq V_i^*$, such that $u \in S$ and that $S \cap V_i \neq \emptyset$. Without loss of generality, we assume that u is a k_i-excess reachable agent in $G[V_i]$, given the choice S, and that u and v are connected by an edge. We observe that there are 2 cases of interest: 1) $v \notin S$ 2) $v \in S$. In the first case, v being external to S increases the count of external neighbors, leading to $k_i + 1 > k_i$ and maintaining 0 reachability. In case 2) both u and v belong to S, therefore the count of internal neighbors is increased by 1. This yields $k_i - 1 \geq 0$, since the exclusion of the case $V_i = V$ implies $k_i \geq 1$. Extending to the case where k_i agents external to V_i are connected to u, we observe that, at worst case, S contains all k_i external agents, yielding $k_i - k_i \geq 0$. Any set of k_i additions is independent of each other, thus not affecting the reachability of the agents in V_i with respect to each other. Additionally, for any agent v belonging to a different community $G[V_{j \neq i}]$, the same reasoning applies from v's perspective, in this way proving the *robustness property*. See Fig. 3 for a graphic representation the two cases.

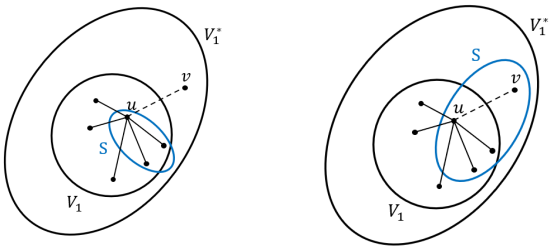


Fig. 3: Example of *Case1* (left) and *Case2* (right) in Proposition 1. With respect to S and selecting the subset V_i, agent u is $3 - 2 = 1$-excess reachable. When considering V_i^*, one agent more is outside S and u becomes 2-excess reachable. In *Case2*, u becomes $3 - 3 = 0$-excess reachable.

b) Isolation property

Consider again the community $G[V_i]$ and k_i external agents connected to $u \in V_i$. For $G[V_i]$ to be isolated, we need to guarantee that the value $\xi_u(t) \in [\xi_{m_i}, \xi_{M_i}]$, respecting the *safety condition*. It is sufficient to show that the median of the values from u's neighbors is $M(\boldsymbol{\xi}^{(u)}(t)) \in [\xi_{m_i}, \xi_{M_i}]$. We prove this in the following. By hypothesis, we know that $d_u^{V_i} \geq d_{min}^{V_i} \geq 2f_i + k_i + 1$. By our model assumptions on the structure of G, any agent $u \in V_i$ can be connected to at most k agents that are external to V_i. Thus, the degree of u with respect to the whole G is $d_u = d_u^{V_i} + k_i$ yielding

$$\begin{aligned} d_u &= d_u^{V_i} + k_i \geq d_{min}^{V_i} + k_i \\ &\geq 2f_i + 2k_i + 1 \geq 2(f_i + k_i) + 1. \end{aligned} \tag{2}$$

The lower bound on d_u in Eq. (2) guarantees that the median operator will not be affected by any value $\xi_v(t) \notin [\xi_{m_i}, \xi_{M_i}]$. This includes both values from legitimate agents that are outside the interval $[\xi_{m_i}, \xi_{M_i}]$ and any value from external malicious agents. This proves isolation and the proposition. ■

C. Proof of Theorem 1

Take a subset $V_i \subseteq V$ for which $G[V_i]$ is a (k_i, f_i)-community with $d_{min}^{V_i} \geq 2f_i + k_i + 1$. From Proposition 1 we know that $G[V_i]$ is isolated and that, the $f_i + 1$ k_i-excess reachable agents in $G[V_i]$ will all be at least 0-excess reachable in G. This means that all of these agents still connect to at least as many agents outside a chosen subset S, *s.t.* $S \cap V_i \neq \emptyset$, as agents inside it. Isolation guarantees that no outliers from agents outside of V_i are selected. This

satisfies the reachability conditions required from Theorem 1 in [1], with respect to subset V_i. Precisely, that result leverages on the fact that $(0, f_i+1)$-robustness of the graph implies that there is always a combination of subsets S_1, S_2 for which enough agents are at least 0-excess reachable. This concludes that MCA on G will lead agents in V_i to reach resilient asymptotic consensus and concludes the proof. ■

Implications – The implications of Theorem 1 and Proposition 1 is twofold. On one hand, these results show that, even if consensus via MCA is not achieved globally, sub-graphs in the network which are connected enough can still agree on a common value. On the other hand, they give sufficient conditions to have a network of (r, s)-excess robust sub-graphs whose agreement to a value will not be hindered by being connected to each other. This is invaluable in many cases where several networks want to communicate with each other, yet still retaining the local information they converged to within themselves, and without compromising it, nor having it corrupted by further malicious agents. Observe that these results do not imply anything about agents that do not belong to a community. In that case consensus may or may not be reached.

VII. Simulation Results

In this section we complement our theoretical findings with simulations, and highlight cases where our conditions on the connectivity and degree are not met, resulting in community consensus failing. We show the successful case of community consensus when both the robustness and degree conditions are met, and two failure scenarios where one of these conditions is violated, respectively.

A. Setup

We start by describing the initial graph G that we then slightly modify to get the three cases. To obtain G, we start from two complete graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ with sizes n_1 and n_2, ($n_1 \geq n_2$ w.l.o.g.), and we set $k_1 = k_2 = k$. Furthermore, we sample each $\xi_u(0)$ such that

$$\xi_u(0) \sim \begin{cases} N(2,1) & \text{if} \quad u \in V_1 \cap L \\ N(30,5) & \text{if} \quad u \in V_2 \cap L \\ 60 & \text{if} \quad u \in F, \end{cases}$$

where $N(\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2. As an example of an effective disruptive behaviour, we assign the same value to all the malicious agents and keep that value constant. We consider three different setups:

- $G[V_1]$ and $G[V_2]$ are (k, f_i)-communities, this is our nominal case;
- At least one of the subgraphs $G[V_i]$ does not respect robustness constraints, that is, it is not (k_i, f_i+1)-excess robust;
- At least one of the subgraphs $G[V_i]$ does not respect the minimum degree condition $d^{V_i}_{min} \geq 2f_i + k_i + 1$.

a) Example1: Constraints Respected

We take $f_1 = 20$, $f_2 = 10$, and $n_1 = 6f_1+3$, $n_2 = 3f_2+5$. We set $k = 2$. It can be verified that each complete graph is $(2, f_i)$-excess robust and the graph G obtained in this way respects the constraints in Theorem 1. Note that the choice of n_i, as well as the parameters of the distributions and the value for the malicious agents, is completely arbitrary, as long as the minimum degree condition $d_{min} = d \geq 2f_i + 2 + 1$ is respected.

b) Example2: Robustness not Respected

We set here $f_1 = 6$, $f_2 = 1$, $n_1 = 2f_1 + 4 = 16$ and $n_2 = 4f_2 + 5 = 9$. In this case we set $k = 1$. By removing a number of edges from the 9-clique, we obtain the sub-graph $G[V_2]$ shown in Fig. 5. This network is not $(1, 2)$-excess robust, and one selection of subsets S_1 and S_2 that breaks the conditions for robustness is the subsets $S_1 = \{u_1, u_2, u_3, u_4, u_5\}$ and $S_2 = \{u_6, u_7, u_8, u_9\}$, highlighted in Fig. 5 by the dashed line. Nevertheless, the minimum degree condition is respected, since every agent in $G[V_2]$ has at least degree 4.

c) Example3: Min Degree not Respected

We set $f_1 = 6$, $f_2 = 3$, $n_1 = 2f_1 + 3 = 15$ and $n_2 = 2f_2 + 5 = 11$ agents each, and set $k = 2$. The minimum degree condition for $G[V_1]$ would be $d^{V_1}_{min} \geq 2f_1 + k + 1 = 15$, which however cannot be respected since every node in $G[V_1]$ has degree $d = 14$. On the other hand, the reader can verify that $G[V_1]$ is $(2, 7)-$excess robust.

d) Code

The code for our experiments (*Phase1.py*) can be found here: `https://bitbucket.org/CrissGava/majorityconsensuscode/src/master/`

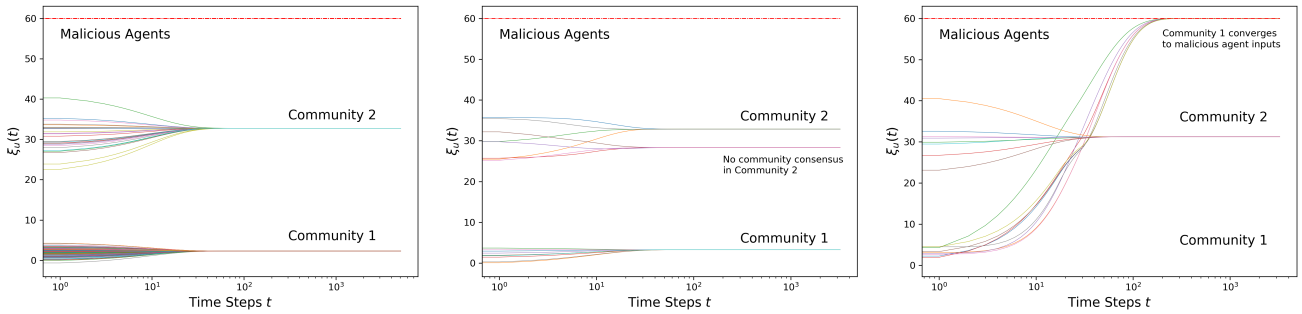
B. Findings

The three plots in Fig. 4a–Fig. 4c show the outcomes of each example. Each line in the plot represents the value of an agent. Note, that the convergence rate is intentionally set to be very slow by setting $\alpha = 0.9$ in the consensus update rule. This is done to provide more insight into the convergence process.

Successful community consensus – When the conditions of Theorem 1 are met, community consensus is reached, as shown in Fig. 4a. At $t = 0$ all the agents start with different values, and they quickly converge within their communities.

Failure due to violation of robustness – Not respecting the robustness condition means that if the malicious agents are positioned strategically in important nodes of the graph, they can prevent information flow between some subsets of the legitimate agents. Consequently, those subsets – depending on their initial values – might converge to different values from each other. This is exactly what happens in $G[V_2]$ in Fig. 4b. Notice, that since the degree condition is respected, all of the values of the legitimate agents stay within the convex hull of their initial values.

Failure due to violation of minimum degree – When the minimum degree condition is not respected in $G[V_i]$, legitimate agents can include values of malicious agents, or legitimate agents from outside $G[V_i]$ in their update rule. Whether this happens or not is dependent on both the initialization of all agents and the placement of the malicious agents. The effects of such inclusion can result in legitimate agents in $G[V_i]$ possibly converging to values outside the



(a) Community Consensus reached (b) Robustness constrain violated (c) Min degree constraint violated

Fig. 4: Plots of our simulation results (cf. Section VII). (a) - *Example1*: Constraints from Theorem 1 are respected, (b) - *Example2*: Robustness constraint is violated: agents in $G[V_2]$ converge to two different values, (c) - *Example3*: Minimum degree constraint is violated. Only in (a) do legitimate agents reach community consensus. The value of 60 belongs to malicious agents and that the x-axis uses logarithmic scale.

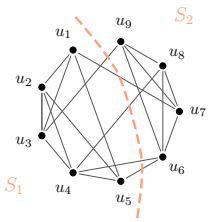


Fig. 5: Graph used to simulate the case where robustness constraints are not met. The dashed line separates V_2 in two subsets of agent: $S_1 = \{u_1, \dots, u_5\}$ and $S_2 = \{u_6, \dots, u_9\}$. Recall that, in this case, it is $f_2 = 1$ and $k_2 = 1$. Therefore it has to be $d_{min}^{V_2} \geq 2f_2 + k_2 + 1 = 4$. In this example, no agent in V_2 is 1-excess reachable under the partition $\{S_1, S_2\}$, thus $G[V_2]$ is not $(1, 2)$-excess robust.

convex hull of their initial values. This is showcased in Fig. 4c, with agents in $G[V_1]$ converging to the values of the malicious agents.

VIII. Practical Use Cases

We argue that Community Consensus can be applied to several real-life use cases. Heterogeneous connectivity in networks is a common occurrence, and many theoretical models account for it, from scale-free, or assortative networks, to different agent centrality measures and clustering criteria. We now present here two use cases

Distributed Sensing – Multiple sensor networks are implemented for various applications: from autonomous vehicle coordination, to energy infrastructure, or seismic and geothermal monitoring. These have steadily evolved to large swarms of micro-sensors and Internet of Things (IoT) [30], creating new challenges in the handling of the big amount of data subject to infrastructure constraints, such as bandwidth limitations and processing power. Different solutions are being explored, from mobile-agent-based Distributed Sensor networks (DSN) and sensor fusion algorithms [31], to Distirbuted Acoustic Sensing (DAS) to map ocean floors [32]. Because of the extensive surface that a DAS application to environmental sensing covers and the extreme diversity of the maritime ecosystem, one might want to segregate the data related to different regions of the ocean floor.

Blockchain Sharding – The blockchain technology is extremely powerful when it comes to approving transactions in a decentralized way and while guaranteeing security. This concept is being used in many applications, from finance to large scale IoT [33]. Nonetheless, it is still computationally expensive and poorly scalable. A solution comes with *sharding*, an approach to allocate different parts of a blockchain transaction to different *shards*, or communities [34], [35]. Shards are connected to each other, so to communicate their result and therefore complete the entire transaction. However, issues like shard invalidation can be caused by malicious agents in an even smaller number than what the entire blockchain would be robust to. Few works focused on optimizing related consensus algorithms yet, and even fewer focused on detailing the robustness of the model given by the network structure. We believe that there is room for exploration and application of our work in these scenarios, reinforcing its versatility and power.

IX. Conclusions

In this work we presented a novel distributed framework, called community consensus, and gave conditions under which the MCA allows agents to reach consensus within their communities, even if some malicious entities are present.

We show the importance of our conditions by providing settings which do not respect said conditions resulting in the agents failing to reach community consensus. This approach finds a place in practical applications and realistic scenarios, where a lower or heterogeneous connectivity is expected, or where it is in fact necessary to maintain regions of a network connected, and at the same time have them able to converge independently of each other.

REFERENCES

[1] H. Zhang and S. Sundaram, "A simple median-based resilient consensus algorithm," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing*, 2012, pp. 1734–1741.

[2] M. Cao, D. Spielman, and A. Morse, "A lower bound on convergence of a distributed network consensus algorithm," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005, pp. 2356–2361.

[3] M. Cao, A. S. Morse, and B. D. Anderson, "Reaching a consensus in a dynamically changing environment: A graphical approach," *SIAM Journal on Control and Optimization*, vol. 47, no. 2, pp. 575–600, 2008.

[4] J. Lorenz, "A stabilization theorem for dynamics of continuous opinions," *Physica A: Statistical Mechanics and its Applications*, vol. 355, no. 1, pp. 217–223, 2005.

[5] M. H. Degroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.

[6] F. Pasqualetti, A. Bicchi, and F. Bullo, "Consensus computation in unreliable networks: A system theoretic approach," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 90–104, 2011.

[7] M. Yemini, A. Nedić, A. J. Goldsmith, and S. Gil, "Characterizing trust and resilience in distributed consensus for cyberphysical systems," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 71–91, 2021.

[8] S. Gil, C. Baykal, and D. Rus, "Resilient multi-agent consensus using wi-fi signals," *IEEE control systems letters*, vol. 3, no. 1, pp. 126–131, 2018.

[9] B. Wang, W. Chen, J. Wang, B. Zhang, Z. Zhang, and X. Qiu, "Cooperative tracking control of multiagent systems: A heterogeneous coupling network and intermittent communication framework," *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4308–4320, 2019.

[10] G. Parlangeli and M. E. Valcher, "On the detection and identification of edge disconnections in a multi-agent consensus network," *arXiv preprint arXiv:2101.06728*, 2021.

[11] F. Mallmann-Trenn, M. Cavorsi, and S. Gil, "Crowd vetting: Rejecting adversaries via collaboration with application to multirobot flocking," *IEEE Trans. Robotics*, vol. 38, no. 1, pp. 5–24, 2022. DOI: 10.1109/TRO.2021.3089033.

[12] J. Lorenz, "Repeated averaging and bounded confidence modeling, analysis and simulation of continuous opinion dynamics," Ph.D. dissertation, Universität Bremen, 2007.

[13] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[14] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[15] M. Cao, A. Morse, and B. Anderson, "Reaching a consensus in a dynamically changing environment: Convergence rates, measurement delays, and asynchronous events," *SIAM J. Control and Optimization*, vol. 47, pp. 601–623, Jan. 2008.

[16] M. Cao, A. S. Morse, and B. D. O. Anderson, "Agreeing asynchronously," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1826–1838, 2008.

[17] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical review letters*, vol. 75, no. 6, p. 1226, 1995.

[18] P. Berenbrink, C. Cooper, C. Gava, *et al.*, *Distributed averaging in population protocols*, 2023. arXiv: 2211.17125.

[19] 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.*, 2003, pp. 482–491.

[20] F. Kuhn, T. Locher, and R. Wattenhofer, "Tight bounds for distributed selection," 2007.

[21] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[22] Y. Chen, L. Su, and J. Xu, "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 1, no. 2, pp. 1–25, 2017.

[23] 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.

[24] R. Guerraoui, S. Rouault, *et al.*, "The hidden vulnerability of distributed learning in byzantium," in *International Conference on Machine Learning*, PMLR, 2018, pp. 3521–3530.

[25] D. Angluin, J. Aspnes, and D. Eisenstat, "A simple population protocol for fast robust approximate majority," *Distributed Computing*, vol. 21, no. 2, pp. 87–102, 2008.

[26] B. Doerr, L. A. Goldberg, L. Minder, T. Sauerwald, and C. Scheideler, "Stabilizing consensus with the power of two choices," 2011.

[27] D. Angluin, M. J. Fischer, and H. Jiang, "Stabilizing consensus in mobile networks," in *Distributed Computing in Sensor Systems*, P. B. Gibbons, T. Abdelzaher, J. Aspnes, and R. Rao, Eds., Springer Berlin Heidelberg, 2006, pp. 37–50.

[28] H. Zhang and S. Sundaram, "Robustness of information diffusion algorithms to locally bounded adversaries," in *2012 American Control Conference (ACC)*, 2012, pp. 5855–5861.

[29] H. Zhang and S. Sundaram, "Robustness of complex networks with implications for consensus and contagion," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 3426–3432.

[30] S. Abbasian Dehkordi, K. Farajzadeh, J. Rezazadeh, R. Farahbakhsh, K. Sandrasegaran, and M. Abbasian Dehkordi, "A survey on data aggregation techniques in iot sensor networks," *Wireless Networks*, vol. 26, pp. 1243–1263, 2020.

[31] "Distributed sensor networks–a review of recent research," *Journal of the Franklin Institute*, vol. 338, no. 6, pp. 655–668, 2001, Distributed Sensor Networks for Real-time Systems with Adaptive C onfiguration.

[32] A. Sladen, D. Rivet, J. P. Ampuero, *et al.*, "Distributed sensing of earthquakes and ocean-solid earth interactions on seafloor telecom cables," *Nature Communications*, vol. 10, p. 5777, Dec. 2019.

[33] X. Cai, S. Geng, J. Zhang, *et al.*, "A sharding scheme-based many-objective optimization algorithm for enhancing security in blockchain-enabled industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7650–7658, 2021.

[34] Y. Liu, J. Liu, M. A. Vaz Salles, *et al.*, "Building blocks of sharding blockchain systems: Concepts, approaches, and open problems," *Computer Science Review*, vol. 46, p. 100 513, 2022.

[35] G. Wang, Z. J. Shi, M. Nixon, and S. Han, "Sok: Sharding on blockchain," in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, 2019, pp. 41–61.