

## MIT Open Access Articles

*Defending non-Bayesian learning against adversarial attacks*

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

**As Published:** <https://doi.org/10.1007/s00446-018-0336-4>

**Publisher:** Springer Berlin Heidelberg

**Persistent URL:** <https://hdl.handle.net/1721.1/131300>

**Version:** Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

**Terms of Use:** Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.



# Defending Non-Bayesian Learning against Adversarial Attacks

Lili Su · Nitin H. Vaidya

Received: date / Accepted: date

**Abstract** This paper addresses the problem of non-Bayesian learning over multi-agent networks, where agents repeatedly collect partially informative observations about an *unknown* state of the world, and try to collaboratively learn the true state out of  $m$  alternatives. We focus on the impact of adversarial agents on the performance of consensus-based non-Bayesian learning, where non-faulty agents combine local learning updates with consensus primitives. In particular, we consider the scenario where an unknown subset of agents suffer Byzantine faults – agents suffering Byzantine faults behave arbitrarily. We propose two learning rules. In our learning rules, each non-faulty agent keeps a local variable which is a stochastic vector over the  $m$  possible states. Entries of this stochastic vector can be viewed as the scores assigned to the corresponding states by that agent. We say a non-faulty agent learns the underlying truth if it assigns one to the true state and zeros to the wrong states asymptotically.

---

This research is supported in part by National Science Foundation award NSF 1421918. Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies or the U.S. government.

A short version of this manuscript [39] has been accepted to appear in the Proceedings of the International Symposium on Distributed Computing (DISC), Sep, 2016

---

L. Su  
Massachusetts Institute of Technology, 32 Vassar St, Cambridge, MA 02139  
Tel.: +180-84-433198  
E-mail: lilisu@mit.edu

N.H. Vaidya  
University of Illinois at Urbana-Champaign, 1308 W. Main St. Urbana, IL 61801

- In our first update rule, each agent updates its local score vector as (up to normalization) the product of (1) the likelihood of the *cumulative* private signals and (2) the weighted geometric average of the score vectors of its incoming neighbors and itself. Under reasonable assumptions on the underlying network structure and the global identifiability of the network, we show that all the non-faulty agents asymptotically learn the true state almost surely.
- We propose a modified variant of our first learning rule whose complexity per iteration per agent is  $O(m^2 n \log n)$ , where  $n$  is the number of agents in the network. In addition, we show that this modified learning rule works under a less restrictive network identifiability condition.

**Keywords** Distributed learning · Byzantine agreement · fault-tolerance · adversary attacks · security

## 1 Introduction

Decentralized hypothesis testing (learning) has received significant amount of attention [5–7, 9–12]. The traditional decentralized detection framework consists of a collection of spatially distributed sensors and a fusion center [9–11]. The sensors independently collect *noisy* signals of the environment state, and send only *summary* of the private signals to the fusion center, where a final decision is made. In the case when the sensors directly send all the private signals, the detection problem can be solved using a centralized scheme. The above framework does not scale well, since each sensor needs to be connected to the fusion center and full reliability of the fusion center is required, which may not be practical as the system scales.

Distributed hypothesis testing in the *absence* of fusion center is considered in [6, 25–27]. In particular, Gale and Kariv [6] studied the distributed hypothesis testing problem in the context of social learning, where fully Bayesian belief update rule is studied. Bayesian update rule is impractical in many applications due to memory and computation constraints of each agent.

To avoid the complexity of Bayesian learning, a non-Bayesian learning framework that combines local learning with distributed consensus was proposed by Jadbabaie et al. [7], and has attracted much attention [14, 18–20, 15, 22, 21, 17]. Jadbabaie et al. [7] considered the general setting where external signals are observed during each iteration of the algorithm execution. Specifically, the belief of each agent is repeatedly updated as the arithmetic mean of its local Bayesian update and the beliefs of its neighbors – combining iterative consensus algorithm with local Bayesian update. It is shown [7] that, under this learning rule, each agent learns the true state almost surely asymptotically. Similar algorithm is proposed in [3]. Comparing to the algorithm in [3], one advantage of the learning rule in [7] is that it incorporates the newly obtained private signals into learning on the fly. The publication of [7] has inspired significant efforts in designing and analyzing non-Bayesian learning rules with a particular focus on refining the fusion strategies and analyzing the (asymptotic and/or finite-time) convergence rates of the refined algorithms [4, 14, 18–20, 15, 22, 21, 17]. In this paper we are particularly interested in the log-linear learning rule, in which, essentially, each agent updates its belief as the geometric average of the local Bayesian update and its neighbors’ beliefs [19, 14, 18, 20, 15, 22, 21, 17]. The log-linear learning rule is shown to converge exponentially fast [15, 14, 20]. Taking an axiomatic approach, the geometric averaging fusion is proved to be the only learning rule satisfying some natural axioms [17]. An optimization-based interpretation of this rule is presented in [20], using dual averaging method with properly chosen proximal functions. Finite-time convergence rates are investigated independently in [18, 15, 21]. Both [18] and [22] consider time-varying networks, with slightly different network models. Specifically, [18] assumes that the union of every consecutive  $B$  networks is strongly connected, while [22] considers random networks.

The prior work implicitly assumes that the networked agents are reliable in the sense that they correctly follow the specified learning rules. However, in some practical multi-agent networks, this assumption may not hold. For example, in social networks, it is possible that some agents are adversarial, and try to prevent the true state from being learned by the good agents. Thus, this paper focuses on the fault-tolerant version of the non-

Bayesian framework proposed in [7]. In particular, we assume that an unknown subset of agents may suffer Byzantine faults. An agent suffering Byzantine fault may not follow the pre-specified algorithms/protocols, and can *misbehave arbitrarily*. For instance, a faulty agent may lie to other agents. In addition, a faulty agent is assumed to have a complete knowledge of the system, including the network topology, the local functions of all the non-faulty agents, the algorithm specification of the non-faulty agents, the execution of the algorithm, the private signals of all the non-faulty agents, and contents of messages the other agents send to each other. The faulty agents may collude to prevent the non-faulty agents from achieving their goal. An alternative fault model, where some agents may unexpectedly cease computing and communicate with each other asynchronously, is considered in our companion work [38]. Recent work [1, 2] consider a closely related problem (multi-agent target tracking) in the presence of adversarial noise rather than the adversarial agents. The Byzantine fault-tolerance problem was introduced by Pease et al. [28] and has attracted intensive attention from researchers [29–32, 34, 35]. Our goal is to design algorithms that enable all the non-faulty agents to learn the underlying true state asymptotically.

The existing non-Bayesian learning algorithms [14, 15, 17–22] are not robust to Byzantine agents, since the malicious messages sent by the Byzantine agents are indiscriminately utilized in the local belief updates. To prevent the network from being completely controlled by the Byzantine agents, some messages filtering mechanism is needed. On the other hand, the incorporation of Byzantine consensus is non-trivial, since the message filtering typically is a function of the exchanged messages, which themselves are “functions” of (i) the malicious behaviors of Byzantine agents (a message may be sent by a Byzantine agent) and (ii) the random private signals collected in the network (a message may contain the information of these signals). Thus, during the information propagation, the Byzantine agents can create arbitrary and unspecified dependency among iterations.

**Contributions:** We propose two learning rules. In our learning rules, each non-faulty agent keeps a local variable which is a stochastic vector over the possible states. Entries of this stochastic vector can be viewed as the scores assigned to the corresponding states by that agent. We say a non-faulty agent learns the underlying truth if it assigns one to the true state and zeros to the wrong states asymptotically.

- We first propose an update rule wherein each agent iteratively updates its score vector as (up to nor-

malization) the product of (1) the likelihood of the *cumulative* private signals and (2) the weighted geometric average of the score vectors of its incoming neighbors and itself (using Byzantine multi-dimensional consensus). In contrast to the existing algorithms [18, 15], where only the *current* private signal is used in the update, our proposed algorithm relies on the *cumulative* private signals. The use of cumulative private signals is counterintuitive at first glance. It turns out that cumulative observations helps us deal with the aforementioned arbitrary and unspecified dependency caused by the Byzantine agents. Under reasonable assumptions on the underlying network structure and the global identifiability of the network, we show that all the non-faulty agents asymptotically agree on the true state almost surely.

- Due to the adoption of Byzantine multi-dimensional consensus, the computation complexity per agent of the first learning rule is high. The network identifiability condition assumed also scales poorly in the number of possible states  $m$ . Observing this, we propose a modification of our first learning rule. The complexity per iteration per agent of the modified rule is  $O(m^2 n \log n)$ , where  $n$  is the number of agents in the network. It requires a less restrictive network identifiability condition. In addition, this improved condition is independent of  $m$ .

**Outline:** The rest of the paper is organized as follows. Section 2 presents the problem formulation. Section 3 briefly reviews existing results on vector Byzantine consensus, and matrix representation. Our first algorithm and its correctness analysis are presented in Section 4. The modified learning rule and its correctness analysis are summarized in Section 5. Section 6 concludes the paper, and discusses possible extensions.

## 2 Problem Formulation

*Network Model:* Our network model is similar to the model used in [8, 34]. We consider a synchronous system. A collection of  $n$  agents/nodes are connected by a *directed* network  $G(\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} = \{1, \dots, n\}$  and  $\mathcal{E}$  is the collection of *directed* edges. For each  $i \in \mathcal{V}$ , let  $\mathcal{I}_i$  denote the set of incoming neighbors of agent  $i$ . Later in this paper, we will characterize sufficient conditions on the network for our learning rules to work. Throughout this paper, we use the terms *agent* and *node* interchangeably.

*Byzantine Fault:* We consider a canonical fault model in distributed computing – the Byzantine fault model [16]. In Byzantine fault model, it is assumed that in

any execution up to  $f$  agents suffer Byzantine faults. For a given execution, let  $\mathcal{F}$  denote the set of Byzantine agents, and  $\mathcal{N}$  denote the set of non-faulty agents. Throughout this paper, we assume that  $f$  and  $n$  satisfy the condition implicitly imposed by the given topology conditions mentioned later. We assume that each non-faulty agent knows  $f$ , but does not know the *actual* number of faulty agents  $|\mathcal{F}|$ .<sup>1</sup> Note that  $|\mathcal{F}| \leq f$  and  $|\mathcal{N}| \geq n - f$  since at most  $f$  agents may fail.

The set of faulty agents may be different across executions, but is fixed within an execution. An agent suffering Byzantine fault may not follow the pre-specified algorithms, and can *misbehave arbitrarily*. For instance, a faulty agent may lie to other agents arbitrarily. But we do assume that a message receiver knows exactly the message’s sender. The Byzantine agents are also assumed to have complete knowledge of system, including the network topology, underlying running algorithm, the states or even the entire history. The faulty agents may collude to prevent the non-faulty agents from achieving their goal [16].

*Observation Model:* Our observation model is identical to the model used in [7, 17, 18]. Let  $\Theta = \{\theta_1, \theta_2, \dots, \theta_m\}$  denote a set of  $m$  environmental states, which we call *hypotheses*. In the  $t$ -th iteration, each agent *independently* obtains a private signal about the environmental state  $\theta^*$ , which is initially unknown to every agent in the network. Each agent  $i$  knows the structure of its private signal, which is represented by a collection of parameterized marginal distributions  $\mathcal{D}^i = \{\ell_i(w_i|\theta) | \theta \in \Theta, w_i \in \mathcal{S}_i\}$ , where  $\ell_i(\cdot|\theta)$  is the distribution of private signal when  $\theta$  is the true state, and  $\mathcal{S}_i$  is the finite private signal space. For each  $\theta \in \Theta$ , and each  $i \in \mathcal{V}$ , the support of  $\ell_i(\cdot|\theta)$  is the whole signal space, i.e.,  $\ell_i(w_i|\theta) > 0, \forall w_i \in \mathcal{S}_i$  and  $\forall \theta \in \Theta$ . Let  $s_t^i$  be the private signal observed by agent  $i$  in iteration  $t$ , and let  $\mathbf{s}_t = \{s_t^1, s_t^2, \dots, s_t^n\}$  be the signal profile at time  $t$  (i.e., signals observed by the agents in iteration  $t$ ). Given an environmental state  $\theta$ , the signal profile  $\mathbf{s}_t$  is generated according to the joint distribution  $\ell_1(s_t^1|\theta) \times \ell_2(s_t^2|\theta) \times \dots \times \ell_n(s_t^n|\theta)$ . In addition, let  $s_{1,t}^i$  be the signal history up to time  $t$  for agent  $i = 1, \dots, n$ , and let  $\mathbf{s}_{1,t} = \{s_{1,t}^1, s_{1,t}^2, \dots, s_{1,t}^n\}$  be the signal profile history up to time  $t$ .

<sup>1</sup> This is because the upper bound  $f$  can be learned via long-time performance statistics, whereas, the actual size of  $\mathcal{F}$  varies across executions, and may be impossible to be predicted in some applications.

### 3 Byzantine Consensus

In this section, we briefly review relevant existing results on Byzantine consensus. Byzantine consensus has attracted significant amount of attention [29, 30, 33, 31, 34, 32, 35]. While the past work mostly focus on scalar inputs, the more general vector (or multi-dimensional) inputs have been studied recently [35, 33, 32]. Complete communication networks are considered in [35, 33], where tight conditions on the number of agents are identified. Incomplete communication networks are studied in [32]. Closer to the non-Bayesian learning problem is the class of *iterative approximate Byzantine consensus algorithms*, where each agent is only allowed to exchange information about its state with its neighbors. In particular, our learning algorithms build upon *Byz-Iter* algorithm proposed in [32] and a simple algorithm proposed in [34] for iterative Byzantine consensus with vector inputs and scalar inputs, respectively, in incomplete networks. A matrix representation of the non-faulty agents' states evolution under *Byz-Iter* algorithm is provided by [32], which also captures the dynamics of the simple algorithm with scalar inputs in [34]. To make this paper self-contained, in this section, we briefly review the algorithm *Byz-Iter* and its matrix representation.

#### 3.1 Algorithm *Byz-Iter* [32]

Algorithm *Byz-Iter* is based on Tverberg's Theorem [36].

**Theorem 1** [36] *Let  $f$  be a nonnegative integer. Let  $Y$  be a multiset containing vectors from  $\mathbb{R}^m$  such that  $|Y| \geq (m+1)f+1$ . There exists a partition  $Y_1, \dots, Y_{f+1}$  of  $Y$  such that  $Y_i$  is nonempty for  $1 \leq i \leq f+1$ , and the intersection of the convex hulls of  $Y_i$ 's are nonempty, i.e.,  $\cap_{i=1}^{f+1} \text{Conv}(Y_i) \neq \emptyset$ , where  $\text{Conv}(Y_i)$  is the convex hull of  $Y_i$  for  $i = 1, \dots, f+1$ .*

The proper partition in Theorem 1, and the points in  $\cap_{i=1}^{f+1} \text{Conv}(Y_i)$ , are referred as *Tverberg partition* of  $Y$  and *Tverberg points* of  $Y$ , respectively.

For convenience of presenting our algorithm in Section 4, we present *Byz-Iter* (described in Algorithm 2) below using *One-Iter* (described in Algorithm 1) as a primitive. The parameter  $\mathbf{x}^i$  passed to *One-Iter* at agent  $i$ , and  $\mathbf{y}^i$  returned by *One-Iter* are both  $m$ -dimensional vectors. Let  $\mathbf{v}^i$  be the state of agent  $i$  that will be iteratively updated, with  $\mathbf{v}_t^i$  being the state at the end of iteration  $t$  and  $\mathbf{v}_0^i$  being the input of agent  $i$ . In each iteration  $t \geq 1$ , a non-faulty agent performs the steps in *One-Iter*. In particular, in the message receiving step, if a message is not received from some neighbor, that

neighbor must be faulty, as the system is synchronous. In this case, the missing message values are set to some default value. Faulty agents may deviate from the algorithm specification arbitrarily. In *Byz-Iter*, the value returned by *One-Iter* at agent  $i$  is assigned to  $\mathbf{v}_t^i$ .

---

**Algorithm 1:** Algorithm *One-Iter* with input  $\mathbf{x}^i$  at agent  $i$

---

- 1  $Z^i \leftarrow \emptyset$ ;
  - 2 Transmit  $\mathbf{x}^i$  on all outgoing links;
  - 3 Receive messages on all incoming links. % These message values form a multiset  $R^i$  of size  $|\mathcal{I}_i|$ . %
  - 4 **for** every  $C \subseteq R^i \cup \{\mathbf{x}^i\}$  such that  $|C| = (m+1)f+1$  **do**
  - 5     add to  $Z^i$  a Tverberg point of multiset  $C$
  - 6 **end**
  - 7 Compute  $\mathbf{y}^i$  as follows:  $\mathbf{y}^i \leftarrow \frac{1}{1+|Z^i|} (\mathbf{x}^i + \sum_{\mathbf{z} \in Z^i} \mathbf{z})$ ;
  - 8 Return  $\mathbf{y}^i$ ;
- 

---

**Algorithm 2:** Algorithm *Byz-Iter* [32]:  $t$ -th iteration at agent  $i$

---

- 1  $\mathbf{v}_t^i \leftarrow \text{One-Iter}(\mathbf{v}_{t-1}^i)$ ;
- 

*Remark 1* Note that for each agent  $i \in \mathcal{N}$ , the computation complexity per iteration is at least

$$\Omega\left(\binom{|\mathcal{I}_i| + 1}{(m+1)f+1}\right) = \Omega\left(\binom{|\mathcal{I}_i| + 1}{(m+1)f+1}\right).$$

In the worst case,  $|\mathcal{I}_i| + 1 = n$ , and

$$\begin{aligned} \Omega\left(\binom{|\mathcal{I}_i| + 1}{(m+1)f+1}\right) &= \Omega\left(\binom{n}{(m+1)f+1}\right) \\ &= \Omega\left(\left(\frac{n}{e}\right)^{(m+1)f+1}\right). \end{aligned}$$

Since our first learning rule is based on Algorithm *Byz-Iter*, the computation complexity of our first proposed algorithm is also high. Nevertheless, our first learning rule contains our main algorithmic ideas. More importantly, this learning rule can be modified such that the computation complexity per iteration per agent is  $O(m^2 n \log n)$ . Specifically, the modified rule adopts the scalar Byzantine consensus instead of the  $m$ -dimensional consensus.

### 3.2 Correctness of Algorithm *Byz-Iter*

We briefly summarize the aspects of correctness proof of Algorithm 2 from [32] that are necessary for our subsequent discussion. By using the Tverberg points in the update of  $\mathbf{v}_t^i$  above, effectively, the extreme message values (that may potentially be sent by faulty agents) are trimmed away. Informally speaking, trimming certain messages can be viewed as ignoring (or removing) incoming links that carry the outliers. [32] shows that the effective communication network thus obtained can be characterized by a “reduced graph” of  $G(\mathcal{V}, \mathcal{E})$ , defined below. It is important to note that the non-faulty agents **do not** know the identity of the faulty agents.

**Definition 1 ( $m$ -dimensional reduced graph)** An  $m$ -dimensional reduced graph  $\mathcal{H}(\mathcal{N}, \mathcal{E}_{\mathcal{F}})$  of  $G(\mathcal{V}, \mathcal{E})$  is obtained by (i) removing all faulty nodes  $\mathcal{F}$ , and all the links incident on the faulty nodes  $\mathcal{F}$ ; and (ii) for each non-faulty node (nodes in  $\mathcal{N}$ ), removing up to  $mf$  additional incoming links.

**Definition 2** A source component in any given  $m$ -dimensional reduced graph is a strongly connected component (of that reduced graph), which does not have any incoming links from outside that component.

It turns out that the effective communication network is potentially time-varying (partly) due to time-varying behavior of faulty nodes. Assumption 1 below states a condition that is sufficient for reaching approximate Byzantine vector consensus using Algorithm 1 [32].

**Assumption 1** Every  $m$ -dimensional reduced graph of  $G(\mathcal{V}, \mathcal{E})$  contains a unique source component.

Let  $\mathcal{C}_m$  be the set of all the  $m$ -dimensional reduced graph of  $G(\mathcal{V}, \mathcal{E})$ . Define  $\chi_m \triangleq |\mathcal{C}_m|$ . Since  $G(\mathcal{V}, \mathcal{E})$  is finite, we have  $\chi_m < \infty$ . Let  $\mathcal{H}_m \in \mathcal{C}_m$  be an  $m$ -dimensional reduced graph of  $G(\mathcal{V}, \mathcal{E})$  with source component  $\mathcal{S}_{\mathcal{H}_m}$ . Define

$$\gamma_m \triangleq \min_{\mathcal{H}_m \in \mathcal{C}_m} |\mathcal{S}_{\mathcal{H}_m}|, \quad (1)$$

i.e.,  $\gamma_m$  is the minimum source component size among all the  $m$ -dimensional reduced graphs. Note that  $\gamma_m \geq 1$  if Assumption 1 holds for a given  $m$ .

**Theorem 2 [32]** Suppose Assumption 1 holds for a given  $m \geq 1$ . Under Algorithm *Byz-Iter*, all the non-faulty agents (agents in  $\mathcal{N}$ ) reach consensus asymptotically, i.e.,  $\lim_{t \rightarrow \infty} |\mathbf{v}_t^i - \mathbf{v}_t^j| = 0, \forall i, j \in \mathcal{N}$ .

The proof of Theorem 2 relies crucially on a matrix representation of the state evolution.

### 3.3 Matrix Representation [32]

Let  $|\mathcal{F}| = \phi$  (thus,  $0 \leq \phi \leq f$ ). Without loss of generality, assume that agents 1 through  $n - \phi$  are non-faulty, and agents  $n - \phi + 1$  to  $n$  are Byzantine.

**Lemma 1 [32]** Suppose Assumption 1 holds for a given  $m \geq 1$ . The state updates performed by the non-faulty agents in the  $t$ -th iteration ( $t \geq 1$ ) can be expressed as

$$\mathbf{v}_t^i = \sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \mathbf{v}_{t-1}^j, \quad (2)$$

where  $\mathbf{A}[t] \in \mathbb{R}^{(n-\phi) \times (n-\phi)}$  is a row stochastic matrix for which there exists an  $m$ -dimensional reduced graph  $\mathcal{H}_m[t]$  with adjacency matrix  $\mathbf{H}_m[t]$  such that  $\mathbf{A}[t] \geq \beta_m \mathbf{H}_m[t]$ , where  $0 < \beta_m \leq 1$  is a constant that depends only on  $G(\mathcal{V}, \mathcal{E})$ .

Let  $\Phi(t, r) \triangleq \mathbf{A}[t] \cdots \mathbf{A}[r]$  for  $1 \leq r \leq t + 1$ . By convention,  $\Phi(t, t) = \mathbf{A}[t]$  and  $\Phi(t, t + 1) = \mathbf{I}$ . Note that  $\Phi(t, r)$  is a backward product. Using prior work on coefficients of ergodicity [13], under Assumption 1, it has been shown [32, 23] that

$$\lim_{t \geq r, t \rightarrow \infty} \Phi(t, r) = \mathbf{1}\pi(r), \quad (3)$$

where  $\pi(r) \in \mathbb{R}^{n-\phi}$  is a row stochastic vector, and  $\mathbf{1}$  is the column vector with each entry being 1. Recall that  $\chi_m$  is the total number of  $m$ -dimensional reduced graphs of  $G(\mathcal{V}, \mathcal{E})$ , and  $\beta_m$  is defined in Lemma 1, and  $\phi \triangleq |\mathcal{F}|$ . The convergence rate in (3) is exponential.

**Theorem 3 [32]** Let  $\nu \triangleq \chi_m(n - \phi)$ . For all  $t \geq r \geq 1$ , it holds that  $|\Phi_{ij}(t, r) - \pi_j(r)| \leq (1 - \beta_m^\nu)^{\lceil \frac{t-r+1}{\nu} \rceil}$ .

Recall that  $\gamma_m$  is defined in (1). The next lemma is a consequence of the results in [32].

**Lemma 2 [32]** For any  $r \geq 1$ , there exists a reduced graph  $\mathcal{H}[r]$  with source component  $\mathcal{S}_r$  such that  $\pi_i(r) \geq \beta_m^{\chi_m(n-\phi)}$  for each  $i \in \mathcal{S}_r$ . In addition,  $|\mathcal{S}_r| \geq \gamma_m$ .

### 3.4 Tight Topological Condition for Scalar Iterative Byzantine Consensus

The above analysis shows that Assumption 1 is sufficient for achieving Byzantine consensus iteratively. For the special case when  $m = 1$ , (i.e., the inputs provided at individual non-faulty agents are scalars) it has been shown [34] that Assumption 1 is also necessary.

**Theorem 4 [34]** For scalar inputs, iterative approximate Byzantine consensus is achievable among non-faulty agents if and only if every 1-dimensional reduced graph of  $G(\mathcal{V}, \mathcal{E})$  contains only one source component.

---

**Algorithm 3:** Algorithm Scalar Byzantine Consensus: iteration  $t \geq 1$  [34]

---

- 1 Transmit  $v^i[t-1]$  on all outgoing links;
  - 2 Receive messages on all incoming links. % These message values  $w_j[t]$  for each  $j \in \mathcal{I}_i$  form a multiset  $R^i[t]$  of size  $|\mathcal{I}_i|$ . %
  - 3 Sort the received values  $w_j[t]$  for each  $j \in \mathcal{I}_i$  in a non-decreasing order;
  - 4 Remove the largest  $f$  values and the smallest  $f$  values. % Denote the set of indices of incoming neighbors whose values have not been removed at iteration  $t$  by  $\mathcal{I}_i^*[t]$ . %
  - 5 Update  $v^i$  as follows:  $v^i[t] \leftarrow \frac{\sum_{j \in \mathcal{I}_i^*[t]} w_j[t] + v^i[t-1]}{1 + |\mathcal{I}_i^*[t]|}$ ;
- 

Moreover, the following simple algorithm (Algorithm 3) works under Assumption 1 when  $m = 1$ .

In addition, it has been show that the dynamic of the non-faulty agents states admits the same matrix representation as in Subsection 3.3 with the reduced graph being 1-dimensional reduced graph defined in Definition 1.

With the above background on Byzantine vector consensus, we are now ready to present our first algorithm and its analysis.

#### 4 Byzantine Fault-Tolerant Non-Bayesian Learning (BFL)

In this section, we present our first learning rule, named Byzantine Fault-Tolerant Non-Bayesian Learning (BFL). In BFL, each non-faulty agent  $i$  keeps a local variable which is a stochastic vector over the possible states, denoted by  $\mu^i \in \mathbb{R}^m$ . Entries of this stochastic vector, i.e.,  $\mu^i(\theta)$  for  $\theta \in \{\theta_1, \dots, \theta_m\}$ , can be viewed as the scores assigned to the corresponding states by that agent. We refer to these stochastic vectors as score vectors.

Since no signals are observed before the execution of an algorithm, the score vector  $\mu^i$  is initially set to be uniform over the set  $\Theta$ , i.e.,  $(\mu_0^i(\theta_1), \mu_0^i(\theta_1), \dots, \mu_0^i(\theta_m))^T = (\frac{1}{m}, \dots, \frac{1}{m})^T$ . Recall that  $\theta^*$  is the true environmental state. We say a non-faulty agent learns the underlying truth  $\theta^*$  if it assigns one to the true state and zeros to the wrong states asymptotically, i.e., for every non-faulty agent  $i \in \mathcal{N}$ ,

$$\lim_{t \rightarrow \infty} \mu_t^i(\theta^*) = 1 \text{ a.s.} \quad (4)$$

where *a.s.* denotes *almost surely*.

BFL modifies the existing geometric averaging update rules [18, 19, 15, 21] to deal with Byzantine agents.

Specifically, in each iteration we use the likelihood of the *cumulative* private signals (instead of the *current* private signals only) to update the local score vectors.

For  $t \geq 1$ , the steps to be performed by agent  $i$  in the  $t$ -th iteration are listed below, where  $\log \mu_{t-1}^i$  means entry-wise taking log of the score vector  $\mu_{t-1}^i$ . Note that faulty agents can deviate from the algorithm specification. The algorithm below uses *One-Iter* presented in the previous section as a primitive. Recall that  $s_{1,t}^i$  is the cumulative private signals up to iteration  $t$ . Since the private signals of a given agent are *i.i.d.*, it holds that  $\ell_i(s_{1,t}^i|\theta) = \prod_{r=1}^t \ell_i(s_r^i|\theta)$ . So  $\ell_i(s_{1,t}^i|\theta)$  can be computed iteratively. In addition, comparing to learning rules in [18, 19, 15, 21], in our BFL, each agent keeps only one additional variable to keep track of the cumulative likelihood, whose update only requires one multiplication operation per state/hypothesis.

---

**Algorithm 4:** BFL: Iteration  $t \geq 1$  at agent  $i$

---

- 1  $\eta_t^i \leftarrow \text{One-Iter}(\log \mu_{t-1}^i)$ ;
  - 2 Observe  $s_t^i$ ;
  - 3 **for**  $\theta \in \Theta$  **do**
  - 4      $\ell_i(s_{1,t}^i|\theta) \leftarrow \ell_i(s_t^i|\theta) \ell_i(s_{1,t-1}^i|\theta)$ ;
  - 5      $\mu_t^i(\theta) \leftarrow \frac{\ell_i(s_{1,t}^i|\theta) \exp(\eta_t^i(\theta))}{\sum_{p=1}^m \ell_i(s_{1,t}^i|\theta_p) \exp(\eta_t^i(\theta_p))}$ ;
  - 6 **end**
- 

*Remark 2* Note that the cumulative likelihood  $\ell_i(s_{1,t}^i|\theta)$  may approach zero, which may cause some implementation issue of BFL. One possible way to resolve this implementation issue is to normalize the cumulative likelihood, i.e.,

$$\hat{\ell}_i(s_{1,t}^i|\theta) \triangleq \frac{\ell_i(s_{1,t}^i|\theta)}{\text{Nor}_t^i}$$

such that  $\sum_{\theta \in \Theta} \hat{\ell}_i(s_{1,t}^i|\theta) = 1$ , where  $\text{Nor}_t^i$  is the normalization constant applied by agent  $i$  at time  $t$ .

The main difference of Algorithm 4 with respect to the algorithms in [18, 19, 15, 21] is that (i) our algorithm uses a Byzantine consensus iteration as a primitive (in line 1), and (ii)  $\ell_i(s_{1,t}^i|\theta)$  used in line 5 is the likelihood for private signals from iteration 1 to  $t$  (the previous algorithms instead use  $\ell_i(s_t^i|\theta)$  here). Observe that the consensus step is being performed on log of the score vector, with the result being stored as  $\eta_t^i$  (in line 1) and used in line 4 to compute the new scores.

Recalling the matrix representation of the *Byz-Iter* algorithm as per Lemma 1, we can write the following

equivalent representation of line 1 of Algorithm 4.

$$\begin{aligned}\eta_t^i(\theta) &= \sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \log \mu_{t-1}^j(\theta) \\ &= \log \prod_{j=1}^{n-\phi} \mu_{t-1}^j(\theta)^{\mathbf{A}_{ij}[t]}, \quad \forall \theta \in \Theta.\end{aligned}\quad (5)$$

where  $\mathbf{A}[t]$  is a row stochastic matrix whose properties are specified in Lemma 1. In addition,  $\mathbf{A}[t]$  can be viewed as the filtering functions of the exchanged messages (score vectors), which are themselves “functions” of (i) the malicious behaviors of Byzantine agents (a message may be sent by a Byzantine agent) and (ii) the random private signals collected in the network (a message may contain the information of these signals). Thus, during the information propagation, the Byzantine agents can create arbitrary and unspecified dependency among the iterations and the aggregated scores. In contrast, the corresponding matrices in [18, 19, 15, 21] do not depend on the exchanged messages.

#### 4.1 Identifiability

In the absence of agent failures [7], for the networked agents to detect the true hypothesis  $\theta^*$ , it is sufficient to assume that  $G(\mathcal{V}, \mathcal{E})$  is strongly connected, and that  $\theta^*$  is globally identifiable. That is, for any  $\theta \neq \theta^*$ , there exists a node  $j \in \mathcal{V}$  such that the Kullback-Leiber divergence between the true marginal  $\ell_j(\cdot|\theta^*)$  and  $\ell_j(\cdot|\theta)$ , denoted by  $D(\ell_j(\cdot|\theta^*)||\ell_j(\cdot|\theta))$ , is nonzero; equivalently,

$$\sum_{j \in \mathcal{V}} D(\ell_j(\cdot|\theta^*)||\ell_j(\cdot|\theta)) \neq 0, \quad (6)$$

where  $D(\ell_j(\cdot|\theta^*)||\ell_j(\cdot|\theta))$  is defined as

$$D(\ell_j(\cdot|\theta^*)||\ell_j(\cdot|\theta)) \triangleq \sum_{w_j \in \mathcal{S}_j} \ell_j(w_j|\theta^*) \log \frac{\ell_j(w_j|\theta^*)}{\ell_j(w_j|\theta)}. \quad (7)$$

Since  $\theta^*$  may change from execution to execution, (6) is required to hold for any choice of  $\theta^*$ . Intuitively speaking, if any pair of states  $\theta_1$  and  $\theta_2$  can be distinguished by at least one agent in the network, then sufficient information exchange over strongly connected network will enable every agent to distinguish  $\theta_1$  and  $\theta_2$ . However, in the presence of Byzantine agents, a stronger global identifiability condition is required as the information exchange is obstructed by the Byzantine agents. The following assumption builds upon Assumption 1.

**Assumption 2** Suppose that Assumption 1 holds for  $m = |\Theta|$ . For any  $\theta \neq \theta^*$ , and for any  $m$ -dimensional reduced graph  $\mathcal{H}$  of  $G(\mathcal{V}, \mathcal{E})$  with  $\mathcal{S}_{\mathcal{H}}$  denoting the unique source component, the following holds

$$\sum_{j \in \mathcal{S}_{\mathcal{H}}} D(\ell_j(\cdot|\theta^*)||\ell_j(\cdot|\theta)) \neq 0. \quad (8)$$

In contrast to (6), where the summation is taken over all the agents in the network, in (8), the summation is taken over agents in the source component only. Intuitively, the condition imposed by Assumption 2 is that all the agents in the source component can detect the true state  $\theta^*$  collaboratively. If iterative consensus is achieved, the accurate scores can be propagated from the source component to every other non-faulty agent in the network.

*Remark 3* We will show later that when Assumption 2 holds, BFL algorithm enables all the non-faulty agents asymptotically assign scores one to the true state  $\theta^*$  almost surely. That is, Assumption 2 is a sufficient condition for a consensus-based non-Bayesian learning algorithm to exist. However, Assumption 2 is not necessary, observing that Assumption 1 (upon which Assumption 2 builds) is not necessary for  $m$ -dimensional Byzantine consensus algorithms to exist, as illustrated by our second learning rule (described later). Nevertheless, BFL contains our main algorithmic and analytical ideas. In addition, BFL provides an alternative learning rule for the failure-free setting (where no fault-tolerant consensus primitives are needed).

#### 4.2 Convergence Results

Our proof parallels the structure of a proof in [18], but with some key differences to take into account our update rule for the score vector.

For any  $\theta_1, \theta_2 \in \Theta$ , and any  $i \in \mathcal{V}$ , define  $\psi_t^i(\theta_1, \theta_2)$  and  $\mathcal{L}_t(\theta_1, \theta_2) \in \mathbb{R}^{n-\phi}$  as follows

$$\psi_t^i(\theta_1, \theta_2) \triangleq \log \frac{\mu_t^i(\theta_1)}{\mu_t^i(\theta_2)}, \quad \mathcal{L}_t^i(\theta_1, \theta_2) \triangleq \log \frac{\ell_i(s_t^i|\theta_1)}{\ell_i(s_t^i|\theta_2)}, \quad (9)$$

where  $\mathcal{L}_t^i(\theta_1, \theta_2)$  is the  $i$ -th entry of  $\mathcal{L}_t(\theta_1, \theta_2)$ . To show Algorithm 4 solves (4), we will show that

$$\psi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty, \quad \forall \theta \neq \theta^*,$$

which implies that

$$\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0, \quad \forall \theta \neq \theta^*, \quad \forall i \in \mathcal{N},$$



i.e., all non-faulty agents asymptotically concentrate their scores on the true hypothesis  $\theta^*$ . We do this by investigating the dynamics of the score vectors which is represented compactly in a matrix form.

For each  $\theta \neq \theta^*$ , and each  $i \in \mathcal{N}$ , we have

$$\begin{aligned} \psi_t^i(\theta, \theta^*) &= \log \frac{\mu_t^i(\theta)}{\mu_t^i(\theta^*)} \\ &\stackrel{(a)}{=} \log \left( \prod_{j=1}^{n-\phi} \left( \frac{\mu_{t-1}^j(\theta)}{\mu_{t-1}^j(\theta^*)} \right)^{\mathbf{A}_{ij}[t]} \times \frac{\ell_i(s_{1,t}^i|\theta)}{\ell_i(s_{1,t}^i|\theta^*)} \right) \\ &= \sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \log \frac{\mu_{t-1}^j(\theta)}{\mu_{t-1}^j(\theta^*)} + \log \frac{\ell_i(s_{1,t}^i|\theta)}{\ell_i(s_{1,t}^i|\theta^*)} \\ &= \sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \psi_{t-1}^j(\theta, \theta^*) + \sum_{r=1}^t \mathcal{L}_r^i(\theta, \theta^*), \quad (10) \end{aligned}$$

where equality (a) follows from (5) and the update of  $\mu^i$  in Algorithm 4, and the last equality follows from (9) and the fact that the private signals are *i.i.d.* for each agent.

Let  $\psi_t(\theta, \theta^*) \in \mathbb{R}^{n-\phi}$  be the vector with the  $i$ -th entry being  $\psi_t^i(\theta, \theta^*)$  for all  $i \in \mathcal{N}$ . Recall that  $\mathcal{L}_r(\theta, \theta^*) \in \mathbb{R}^{n-\phi}$  is the vector that stacks all  $\mathcal{L}_r^i(\theta, \theta^*)$  for  $i \in \mathcal{N}$ . The evolution of  $\psi(\theta, \theta^*)$  can be compactly written as

$$\psi_t(\theta, \theta^*) = \mathbf{A}[t] \psi_{t-1}(\theta, \theta^*) + \sum_{r=1}^t \mathcal{L}_r(\theta, \theta^*). \quad (11)$$

Expanding (11), we get

$$\begin{aligned} \psi_t(\theta, \theta^*) &= \Phi(t, 1) \psi_0(\theta, \theta^*) \\ &\quad + \sum_{r=1}^t \Phi(t, r+1) \sum_{k=1}^r \mathcal{L}_k(\theta, \theta^*). \quad (12) \end{aligned}$$

For each  $\theta \in \Theta$  and  $i \in \mathcal{V}$ , define  $H_i(\theta, \theta^*) \in \mathbb{R}^{n-\phi}$  as

$$\begin{aligned} H_i(\theta, \theta^*) &\triangleq \sum_{w_i \in \mathcal{S}_i} \ell_i(w_i|\theta^*) \log \frac{\ell_i(w_i|\theta)}{\ell_i(w_i|\theta^*)} \\ &= -D(\ell_i(\cdot|\theta^*) \parallel \ell_i(\cdot|\theta)) \quad \text{by (7)} \\ &\leq 0. \quad (13) \end{aligned}$$

Let  $\mathcal{H} \in \mathcal{C}$  be an arbitrary reduced graph with source component  $\mathcal{S}_{\mathcal{H}}$ . Define  $C_0$  and  $C_1$  as

$$-C_0 \triangleq \min_{i \in \mathcal{V}} \min_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \min_{w_i \in \mathcal{S}_i} \left( \log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)} \right), \quad (14)$$

$$C_1 \triangleq \min_{\mathcal{H} \in \mathcal{C}} \min_{\theta, \theta^* \in \Theta; \theta \neq \theta^*} \sum_{i \in \mathcal{S}_{\mathcal{H}}} D(\ell_i(\cdot|\theta^*) \parallel \ell_i(\cdot|\theta)). \quad (15)$$

The constant  $C_0$  serves as an universal upper bound on  $|\log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)}|$  for all choices of  $\theta_1$  and  $\theta_2$ , all signals

and all agents. Intuitively, the constant  $C_1$  is the minimal detection capability of the source component under Assumption 2.

Due to  $|\Theta| = m < \infty$  and  $|\mathcal{S}_i| < \infty$  for each  $i \in \mathcal{N}$ , we know that  $C_0 < \infty$ . Besides, it is easy to see that  $-C_0 \leq 0$  (thus,  $C_0 \geq 0$ ). In addition, under Assumption 2, we have  $C_1 > 0$ .

Now we present a key lemma for our main theorem.

**Lemma 3** *Under Assumption 2, for any  $\theta \neq \theta^*$ , it holds that*

$$\begin{aligned} \frac{1}{t^2} \sum_{r=1}^t \left( \sum_{j=1}^{n-\phi} \Phi_{ij}(t, r+1) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \right. \\ \left. - r \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*) \right) \xrightarrow{\text{a.s.}} 0. \end{aligned}$$

As it can be seen later, the proof of Lemma 3 is different from the analogous lemma in [18].

**Theorem 5** *When Assumption 2 holds, each non-faulty agent  $i \in \mathcal{N}$  concentrates its score on the true hypothesis  $\theta^*$  almost surely, i.e.,  $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0$  for all  $\theta \neq \theta^*$ .*

*Proof* Consider any  $\theta \neq \theta^*$ . Recall from (12) that

$$\begin{aligned} \psi_t(\theta, \theta^*) &= \Phi(t, 1) \psi_0(\theta, \theta^*) \\ &\quad + \sum_{r=1}^t \Phi(t, r+1) \sum_{k=1}^r \mathcal{L}_k(\theta, \theta^*) \\ &= \sum_{r=1}^t \Phi(t, r+1) \sum_{k=1}^r \mathcal{L}_k(\theta, \theta^*). \end{aligned}$$

The last equality holds as  $\mu_0^i$  is uniform, and  $\psi_0^i(\theta, \theta^*) = 0$  for each  $i \in \mathcal{N}$ .

Since the supports of  $\ell_i(\cdot|\theta)$  and  $\ell_i(\cdot|\theta^*)$  are the whole signal space  $\mathcal{S}_i$ , which are finite, for each agent  $i \in \mathcal{N}$ , it holds that  $\left| \frac{\ell_i(w_i|\theta)}{\ell_i(w_i|\theta^*)} \right| < \infty$  for each  $w_i \in \mathcal{S}_i$ , and

$$0 \geq H_i(\theta, \theta^*) \geq \min_{w_i \in \mathcal{S}_i} \left( \log \frac{\ell_i(w_i|\theta)}{\ell_i(w_i|\theta^*)} \right) \geq -C_0 > -\infty. \quad (16)$$

By (16), we know that  $|\sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*)| \leq C_0 < \infty$ . Due to the finiteness of  $\sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*)$ , we are able to add to and subtract  $r \mathbf{1} \sum_{j=1}^{n-\phi} \pi_j(r+1)$

1)  $H_j(\theta, \theta^*)$  from (12). We get

$$\begin{aligned} \psi_t(\theta, \theta^*) &= \sum_{r=1}^t \left( \Phi(t, r+1) \sum_{k=1}^r \mathcal{L}_k(\theta, \theta^*) \right. \\ &\quad \left. - r \mathbf{1} \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*) \right) \\ &\quad + \sum_{r=1}^t r \mathbf{1} \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*). \end{aligned} \quad (17)$$

For each  $i \in \mathcal{N}$ , we have

$$\begin{aligned} \psi_t^i(\theta, \theta^*) &= \sum_{r=1}^t \left( \sum_{j=1}^{n-\phi} \Phi_{ij}(t, r+1) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \right. \\ &\quad \left. - r \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*) \right) \\ &\quad + \sum_{r=1}^t r \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*). \end{aligned} \quad (18)$$

To show  $\lim_{t \rightarrow \infty} \mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0$  for  $\theta \neq \theta^*$ , it is enough to show  $\psi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$ . Our convergence proof has similar structure as the analysis in [18]. From Lemma 3, we know that

$$\begin{aligned} \frac{1}{t^2} \sum_{r=1}^t \left( \sum_{j=1}^{n-\phi} \Phi_{ij}(t, r+1) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \right. \\ \left. - r \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*) \right) \xrightarrow{\text{a.s.}} 0. \end{aligned} \quad (19)$$

Next we show that the second term of the right hand side of (18) decreases quadratically in  $t$ .

$$\begin{aligned} &\sum_{r=1}^t r \sum_{j=1}^{n-\phi} \pi_j(r+1) H_j(\theta, \theta^*) \\ &\leq \sum_{r=1}^t r \sum_{j \in \mathcal{S}_r} \pi_j(r+1) H_j(\theta, \theta^*) \quad \text{by (13)} \\ &\leq \sum_{r=1}^t r \beta^{\chi(n-\phi)} \sum_{j \in \mathcal{S}_r} H_j(\theta, \theta^*) \quad \text{by Lemma 2} \\ &\leq - \sum_{r=1}^t r \beta_m^{\chi(n-\phi)} C_1 \quad \text{by (15) and (13)} \\ &\leq - \frac{t^2}{2} \beta_m^{\chi(n-\phi)} C_1. \end{aligned} \quad (20)$$

Therefore, by (18), (19) and (20), almost surely, the following hold

$$\lim_{t \rightarrow \infty} \frac{1}{t^2} \psi_t^i(\theta, \theta^*) \leq - \frac{1}{2} \beta_m^{\chi(n-\phi)} C_1.$$

Therefore, we have  $\psi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$  and  $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0$  for  $i \in \mathcal{N}$  and  $\theta \neq \theta^*$ , proving Theorem 5.

*Remark 4* From the proof of Theorem 5, we know that the asymptotic convergence rate of the local variable  $\mu^i$  is  $O(\exp(-Ct^2))$ , where  $C$  is some constant. At first glance, it seems that this asymptotic convergence rate directly contradicts Stein's Lemma which states that the best possible error decay rate w.r.t. the posterior distribution is  $O(\exp(-\hat{C}t))$ , where  $\hat{C}$  is a function of the KL-divergence of two distributions. However, we need to emphasize the fact that the local variable  $\mu^i$  used in our algorithm is NOT the posterior distribution. To see this, let us consider a special setting when  $n = 1$  and the agent is non-faulty – the standard centralized hypothesis testing problem. According to Algorithm 4, the score vector  $\mu$  is updated as

$$\mu_t(\theta) \leftarrow \frac{\ell(s_{1,t}|\theta) \mu_{t-1}(\theta)}{\sum_{p=1}^m \ell(s_{1,t}^i|\theta_p) \mu_{t-1}(\theta_p)},$$

for all  $\theta \in \Theta = \{\theta_1, \dots, \theta_m\}$ , where

$$\ell(s_{1,t}|\theta) = \prod_{r=1}^t \ell(s_r|\theta)$$

is the likelihood of all the signals  $s_r$  ( $r = 1, \dots, t$ ) collected up to time  $t$ . In contrast, the posterior distribution w.r.t. the cumulative signals  $s_r$  ( $r = 1, \dots, t$ ), denoted by  $\mathbb{P}\{\theta | s_{1,t}\}$ , is defined as

$$\begin{aligned} \mathbb{P}\{\theta | s_{1,t}\} &= \frac{\mathbb{P}\{\theta, s_{1,t}\}}{\mathbb{P}\{s_{1,t}\}} = \frac{\mathbb{P}_\Theta(\theta) \mathbb{P}\{s_{1,t} | \theta\}}{\mathbb{P}\{s_{1,t}\}} \\ &= \frac{\mathbb{P}_\Theta(\theta) \prod_{r=1}^t \ell(s_r|\theta)}{\sum_{\theta_p \in \Theta} \mathbb{P}_\Theta(\theta_p) \prod_{r=1}^t \ell(s_r|\theta_p)}, \end{aligned}$$

where  $\mathbb{P}_\Theta$  is the fixed prior without seeing any signals. Thus, by definition,

$$\mathbb{P}\{\theta | s_{1,t}\} \neq \mu_t(\theta).$$

Note that if we replace the cumulative likelihood by the average of the likelihood of the private signals collected so far, i.e.,

$$\mu_t(\theta) \leftarrow \frac{(\ell(s_{1,t}|\theta))^{\frac{1}{t}} \mu_{t-1}(\theta)}{\sum_{p=1}^m \ell(s_{1,t}^i|\theta_p)^{\frac{1}{t}} \mu_{t-1}(\theta_p)},$$

we are able to obtain the asymptotic rate  $O(\exp(-Ct))$  (using the derivation in the proof of Lemma 3) which is similar to that in the failure-free setting [15, 14, 20]. That is, using cumulative likelihood may not lead to faster convergence rate. Our interests in cumulative likelihood (as well as the averaging version) lies in its “robustness” to adversarial attacks when combined with Byzantine consensus iteration.

We now prove our key lemma – Lemma 3. Henceforth, for ease of exposition, we drop the subscript of  $\beta$ .

*Proof (Proof of Lemma 3)* By (9), we have

$$\begin{aligned} |\mathcal{L}_r^i(\theta, \theta^*)| &= \left| \log \frac{\ell_i(s_t^i|\theta)}{\ell_i(s_t^i|\theta^*)} \right| \\ &\leq \max_{i \in \mathcal{V}} \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \left| \log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)} \right|. \end{aligned}$$

Note that

$$\begin{aligned} &\max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \left| \log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)} \right| \\ &= \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)}. \end{aligned} \quad (21)$$

We prove (21) in Appendix A. Thus, we can rewrite the upper bound of  $|\mathcal{L}_r^i(\theta, \theta^*)|$  as follows.

$$\begin{aligned} |\mathcal{L}_r^i(\theta, \theta^*)| &\leq \max_{i \in \mathcal{V}} \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \left| \log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)} \right| \\ &= \max_{i \in \mathcal{V}} \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \log \frac{\ell_i(w_i|\theta_1)}{\ell_i(w_i|\theta_2)} \\ &= \max_{i \in \mathcal{V}} \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} -\log \frac{\ell_i(w_i|\theta_2)}{\ell_i(w_i|\theta_1)} \\ &= -\min_{i \in \mathcal{V}} \min_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \min_{w_i \in \mathcal{S}_i} \log \frac{\ell_i(w_i|\theta_2)}{\ell_i(w_i|\theta_1)} \\ &= -(-C_0) = C_0 < \infty. \end{aligned} \quad (22)$$

Thus, adding to and subtracting  $\frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*)$  from the first term on the right hand side of (18), we can get

$$\begin{aligned} &\frac{1}{t^2} \sum_{r=1}^t \left( \sum_{j=1}^{n-\phi} \Phi_{ij}(t, r+1) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \right. \\ &\quad \left. - \pi_j(r+1) r \sum_{j=1}^{n-\phi} H_j(\theta, \theta^*) \right) \\ &= \frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} (\Phi_{ij}(t, r+1) - \pi_j(r+1)) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \\ &\quad + \frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - r H_j(\theta, \theta^*) \right). \end{aligned} \quad (23)$$

For the first term of the right hand side of (23), we have

$$\begin{aligned} &\frac{1}{t^2} \left| \sum_{r=1}^t \sum_{j=1}^{n-\phi} (\Phi_{ij}(t, r+1) - \pi_j(r+1)) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \right| \\ &\leq \frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} |\Phi_{ij}(t, r+1) - \pi_j(r+1)| \sum_{k=1}^r |\mathcal{L}_k^j(\theta, \theta^*)| \\ &\leq \frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} |\Phi_{ij}(t, r+1) - \pi_j(r+1)| r C_0 \quad \text{by (22)} \\ &\leq \frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} (1 - \beta^\nu)^{\lceil \frac{t-r}{\nu} \rceil} r C_0 \quad \text{by Theorem 3} \\ &\leq \frac{1}{t^2} (t(n-\phi) C_0) \sum_{r=1}^t (1 - \beta^\nu)^{\lceil \frac{t-r}{\nu} \rceil} \\ &\leq \frac{(n-\phi) C_0}{(1 - \beta^\nu)(1 - (1 - \beta^\nu)^{\frac{1}{\nu}}) t}. \end{aligned} \quad (24)$$

Thus, for every sample path, we have

$$\frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} (\Phi_{ij}(t, r+1) - \pi_j(r+1)) \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) \rightarrow 0.$$

For the second term of the right hand side of (23), we will show that

$$\frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - r H_j(\theta, \theta^*) \right) \xrightarrow{\text{a.s.}} 0,$$

i.e., almost surely for any  $\epsilon > 0$  there exists sufficiently large  $t(\epsilon)$  such that  $\forall t \geq t(\epsilon)$ ,

$$\frac{1}{t^2} \left| \sum_{r=1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - r H_j(\theta, \theta^*) \right) \right| \leq \epsilon. \quad (25)$$

We prove this by dividing  $r$  into two ranges  $\{1, \dots, \sqrt{t}\}$  and  $\{\sqrt{t}+1, \dots, t\}$ , i.e.,

$$\begin{aligned} &\frac{1}{t^2} \sum_{r=1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - r H_j(\theta, \theta^*) \right) \\ &= \frac{1}{t^2} \sum_{r=1}^{\sqrt{t}} \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - r H_j(\theta, \theta^*) \right) \\ &\quad + \frac{1}{t^2} \sum_{r=\sqrt{t}+1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - r H_j(\theta, \theta^*) \right). \end{aligned} \quad (26)$$

For the first term of the right hand side of (26), we have

$$\begin{aligned}
& \frac{1}{t^2} \left| \sum_{r=1}^{\sqrt{t}} \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - rH_j(\theta, \theta^*) \right) \right| \\
& \leq \frac{1}{t^2} \sum_{r=1}^{\sqrt{t}} \sum_{j=1}^{n-\phi} \pi_j(r+1) (2rC_0) \quad \text{by (13) and (22)} \\
& = \frac{1}{t^2} (2C_0) \sum_{r=1}^{\sqrt{t}} r \\
& \leq C_0 \left( \frac{1}{t} + \frac{1}{t^{\frac{3}{2}}} \right).
\end{aligned}$$

Thus, there exists  $t_1(\epsilon)$  such that for all  $t \geq t_1(\epsilon)$ , it holds that

$$\frac{1}{t^2} \left| \sum_{r=1}^{\sqrt{t}} \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - rH_j(\theta, \theta^*) \right) \right| \leq \frac{\epsilon}{2}.$$

For the second term of the right hand side of (26), we have

$$\begin{aligned}
& \frac{1}{t^2} \sum_{r=\sqrt{t}+1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - rH_j(\theta, \theta^*) \right) \\
& = \frac{1}{t} \sum_{r=\sqrt{t}+1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \frac{1}{t} \sum_{k=1}^r \left( \mathcal{L}_k^j(\theta, \theta^*) - H_j(\theta, \theta^*) \right)
\end{aligned}$$

Since  $\mathcal{L}_k^j(\theta, \theta^*)$ 's are i.i.d., from Strong LLN, we know that  $\frac{1}{r} \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - H_j(\theta, \theta^*) \xrightarrow{\text{a.s.}} 0$  as  $r \rightarrow \infty$  for any  $j$  and any  $\theta$ . That is, with probability 1, the sample path of  $\left\{ \mathcal{L}_k^j(\theta, \theta^*) \right\}_{k=1}^\infty$  for any  $j$  and any  $\theta$  converges. Now, focus on each convergent sample path. For sufficiently large  $r(\epsilon)$ , it holds that for any  $r \geq r(\epsilon)$ ,

$$\left| \frac{1}{r} \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - H_j(\theta, \theta^*) \right| \leq \frac{\epsilon}{2}.$$

Recall that  $r \geq \sqrt{t}$ . Thus, we know that there exists sufficiently large  $t_2(\epsilon)$  such that  $\forall t \geq t_2(\epsilon)$ ,  $r \geq \sqrt{t}$  is large enough and

$$\left| \frac{1}{r} \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - H_j(\theta, \theta^*) \right| \leq \frac{\epsilon}{2}.$$

Then, we have  $\forall t \geq t_2(\epsilon)$ ,

$$\begin{aligned}
& \frac{1}{t^2} \left| \sum_{r=\sqrt{t}+1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - rH_j(\theta, \theta^*) \right) \right| \\
& = \frac{1}{t} \sum_{r=\sqrt{t}+1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \frac{r}{t} \left| \frac{1}{r} \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - H_j(\theta, \theta^*) \right| \\
& \leq \frac{1}{t} \sum_{r=\sqrt{t}+1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \frac{r}{t} \frac{\epsilon}{2} \\
& = \frac{1}{t} \sum_{r=\sqrt{t}+1}^t \frac{r}{t} \frac{\epsilon}{2} = \frac{\epsilon}{2} \frac{1}{t^2} \sum_{r=\sqrt{t}+1}^t r \\
& = \frac{\epsilon}{4} \frac{1}{t^2} (t^2 - \sqrt{t}) \leq \frac{\epsilon}{2}.
\end{aligned}$$

Therefore,  $\forall \epsilon > 0$ , there exists  $\max\{t_1(\epsilon), t_2(\epsilon)\}$ , such that for any  $t \geq \max\{t_1(\epsilon), t_2(\epsilon)\}$ ,

$$\frac{1}{t^2} \left| \sum_{r=1}^t \sum_{j=1}^{n-\phi} \pi_j(r+1) \left( \sum_{k=1}^r \mathcal{L}_k^j(\theta, \theta^*) - rH_j(\theta, \theta^*) \right) \right| \leq \epsilon,$$

for every convergent sample path. In addition, we know a sample path is convergent with probability 1. Thus (25) holds almost surely.

Therefore, Lemma 3 is proved.

## 5 Modified BFL

To reduce the computation complexity per iteration in general, and to identify a less restrictive global identifiability condition, we propose a modification of our first learning rule.

We decompose the  $m$ -ary hypothesis testing problem into  $m(m-1)$  (ordered) binary hypothesis testing problems. For each pair of hypotheses  $\theta_1$  and  $\theta_2$ , each non-faulty agent updates the likelihood ratio of  $\theta_1$  over  $\theta_2$  as follows. Let  $r_t^i(\theta_1, \theta_2)$  be the log likelihood ratio of  $\theta_1$  over  $\theta_2$  kept by agent  $i$  at the end of iteration  $t$ . Our modified learning rule applies consensus procedures to log likelihood ratio, i.e.,  $r_t^i(\theta_1, \theta_2)$ , which is a scalar. For Algorithm 5, we only require scalar iterative Byzantine (approximate) consensus among the non-faulty agents to be achievable.

**Assumption 3** Suppose that every 1-dimensional reduced graph of  $G(\mathcal{V}, \mathcal{E})$  contains only one source component. For any  $\theta \neq \theta^*$ , and for any 1-dimensional reduced graph  $\mathcal{H}_1$  of  $G(\mathcal{V}, \mathcal{E})$  with  $\mathcal{S}_{\mathcal{H}_1}$  denoting the unique source component, the following holds

$$\sum_{j \in \mathcal{S}_{\mathcal{H}_1}} D(\ell_j(\cdot | \theta^*) \| \ell_j(\cdot | \theta)) \neq 0. \quad (27)$$

**Algorithm 5:** Pairwise Learning

---

```

1 Initialization: for  $\theta_1, \theta_2 \in \Theta$ , and  $\theta_1 \neq \theta_2$  do
2    $r_0^i(\theta_1, \theta_2) \leftarrow 0$ ;
3 end
4 while  $t \geq 1$  do
5   for  $\theta_1, \theta_2 \in \Theta$ , and  $\theta_1 \neq \theta_2$  do
6     Transmit current score vector  $r_{t-1}^i(\theta_1, \theta_2)$ 
       on all outgoing edges;
7     Wait until a private signal  $s_t^i$  is observed
       and log likelihood ratios  $\tilde{r}_{t-1}^j(\theta_1, \theta_2)$  are
       received from all incoming neighbors  $\mathcal{I}_i$ ;
8     Sort the received log likelihood ratios
        $\tilde{r}_{t-1}^j(\theta_1, \theta_2)$  in a non-decreasing order, and
       remove the smallest  $f$  values and the
       largest  $f$  values. % Denote the set of indices
       of incoming neighbors whose ratios have not
       been removed at iteration  $t$  by  $\mathcal{I}_i^*[t]$ .%
9      $r_t^i(\theta_1, \theta_2) \leftarrow$ 
        $\frac{\sum_{j \in \mathcal{I}_i^*[t]} \tilde{r}_{t-1}^j(\theta_1, \theta_2) + r_{t-1}^i(\theta_1, \theta_2)}{|\mathcal{I}_i^*[t]| + 1} + \log \frac{\ell_i(s_{1,t}^i | \theta_1)}{\ell_i(s_{1,t}^i | \theta_2)}$ .
10   end
11 end

```

---

For each iteration, the computation complexity per agent (non-faulty) can be calculated as follows. The cost-dominant procedure in each iteration is sorting the received log likelihood ratios, which takes  $O(n \log n)$  operations. In total, we have  $m(m-1)$  order pairs of hypotheses. Thus, the total computation per agent per iteration is  $O(m^2 n \log n)$ .

We show in Theorem 6 that the underlying truth  $\theta^*$  satisfies an interesting property, and later in Proposition 1 that  $\theta^*$  is the only state in  $\Theta = \{\theta_1, \dots, \theta_m\}$  satisfies this property.

**Theorem 6** *Suppose Assumption 3 holds. Under Algorithm 5, for any  $\theta \neq \theta^*$ , the following holds:*

$$r_t^i(\theta^*, \theta) \xrightarrow{\text{a.s.}} +\infty, \text{ and } r_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty.$$

*Proof* By [24], we know that for each pair of hypotheses  $\theta_1$  and  $\theta_2$ , there exists a row-stochastic matrix  $\mathbf{M}^{1,2}[t] \in \mathbb{R}^{(n-\phi) \times (n-\phi)}$  such that

$$r_t^i(\theta_1, \theta_2) = \sum_{j=1}^{n-\phi} \mathbf{M}_{ij}^{1,2}[t] r_{t-1}^j(\theta_1, \theta_2) + \log \frac{\ell_i(s_{1,t}^i | \theta_1)}{\ell_i(s_{1,t}^i | \theta_2)}. \quad (28)$$

Note that matrix  $\mathbf{M}^{1,2}$  depends on the choice of hypotheses  $\theta_1$  and  $\theta_2$ .

For a given pair of hypotheses  $\theta_1$  and  $\theta_2$ , let  $\mathbf{r}_t(\theta_1, \theta_2) \in \mathbb{R}^{n-\phi}$  be the vector that stacks  $r_t^i(\theta_1, \theta_2)$ . The evolution

of  $\mathbf{r}(\theta_1, \theta_2)$  can be compactly written as

$$\begin{aligned} \mathbf{r}_t(\theta_1, \theta_2) &= \mathbf{M}^{1,2}[t] \mathbf{r}_{t-1}(\theta_1, \theta_2) + \sum_{r=1}^t \mathcal{L}_r(\theta_1, \theta_2) \\ &= \sum_{r=1}^t \Phi^{1,2}(t, r+1) \sum_{k=1}^r \mathcal{L}_k(\theta_1, \theta_2), \end{aligned} \quad (29)$$

where  $\Phi^{1,2}(t, r+1) \triangleq \mathbf{M}^{1,2}[t] \mathbf{M}^{1,2}[t-1] \dots \mathbf{M}^{1,2}[r+1]$  for  $r \leq t$ ,  $\Phi^{1,2}(t, t) \triangleq \mathbf{M}^{1,2}[t]$  and  $\Phi^{1,2}(t, t+1) \triangleq \mathbf{I}$ . We do the analysis for each pair of  $\theta_1$  and  $\theta_2$  separately.

The remaining proof is identical to the proof of Theorem 5, and is omitted.

The true hypothesis  $\theta^*$  is the only one in  $\Theta$  satisfies the property stated in Theorem 6.

**Proposition 1** *Suppose there exists  $\tilde{\theta} \in \Theta$  such that for any  $\theta \neq \tilde{\theta}$ , it holds that  $r_t^i(\tilde{\theta}, \theta) \xrightarrow{\text{a.s.}} +\infty$ , and  $r_t^i(\theta, \tilde{\theta}) \xrightarrow{\text{a.s.}} -\infty$ . Then  $\tilde{\theta} = \theta^*$ .*

*Proof* We prove this proposition by contradiction. Suppose there exists  $\tilde{\theta} \neq \theta^* \in \Theta$  such that for any  $\theta \neq \tilde{\theta}$ , it holds that  $r_t^i(\tilde{\theta}, \theta) \xrightarrow{\text{a.s.}} +\infty$ , and  $r_t^i(\theta, \tilde{\theta}) \xrightarrow{\text{a.s.}} -\infty$ . Then we know that  $r_t^i(\tilde{\theta}, \theta^*) \xrightarrow{\text{a.s.}} +\infty$  and  $r_t^i(\theta^*, \tilde{\theta}) \xrightarrow{\text{a.s.}} -\infty$ , contradicting Theorem 6. Thus, Proposition 1 is true.

## 6 Conclusion

This paper addresses the problem of consensus-based non-Bayesian learning over multi-agent networks when an unknown subset of agents may be adversarial (Byzantine). We propose two learning rules. In our first update rule, each agent updates its score vector as (up to normalization) the product of (1) the likelihood of the cumulative private signals and (2) the weighted geometric average of the score vectors of its incoming neighbors and itself. Under reasonable assumptions on the underlying network structure and the global identifiability of the network, we show that all the non-faulty agents asymptotically agree on the true state almost surely. Although the asymptotic convergence rate is shown to be double exponential, the physical meaning of this rate might be limited. In fact, using cumulative likelihood might make higher moments very large and affect the concentration of the “score” vectors maintained by the non-faulty agents.

Throughout this paper, we assume that consensus among non-faulty agents needs to be achieved. Although this is necessary for the family of consensus-based algorithms (by definition), this is not the case for the non-faulty agents to collaboratively learn the true state in

general. Indeed, there is a tradeoff between the capability of the network to reach consensus and the tight condition of the network detectability. For instance, if the network is disconnected, then information cannot be propagated across the connected components. Thus, the non-faulty agents in each connected component have to be able to learn the true state. We leave investigating the above tradeoff as future work.

## References

1. Shahrampour, Shahin and Jadbabaie, Ali. An online optimization approach for multi-agent tracking of dynamic parameters in the presence of adversarial noise. *American Control Conference (ACC)*, IEEE, 2017.
2. Bedi, Amrit Singh and Sarma, Paban and Rajawat, Ketan. Adversarial Multi-Agent Target Tracking with Inexact Online Gradient Descent. *arXiv preprint arXiv:1710.05133*, 2017.
3. R. Olfati-Saber, E. Franco, E. Frazzoli and J. Shamma. Belief consensus and distributed hypothesis testing in sensor networks. *Networked Embedded Sensing and Control*, Springer, 2006.
4. M. A. Rahimian and A. L. Jadbabaie. Learning without Recall: A Case for Log-Linear Learning. *IFAC-PapersOnLine*, Volume 48, Issue 22, 2015, pages 46-51, ISSN 2405-8963, <http://dx.doi.org/10.1016/j.ifacol.2015.10.305>.
5. J.-F. Chamberland and V. V. Veeravalli. Decentralized detection in sensor networks. *IEEE Transactions on Signal Processing*, 2003.
6. D. Gale and S. Kariv. Bayesian learning in social networks. *Games and Economic Behavior*, 2003.
7. A. Jadbabaie, P. Molavi, A. Sandroni, and A. Tahbaz-Salehi. Non-bayesian social learning. *Games and Economic Behavior*, 2012.
8. L. Su and N. H. Vaidya. Reaching approximate byzantine consensus with multi-hop communication. In *Proceedings of International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS)*, Aug. 2015.
9. J. Tsitsiklis. Decentralized detection by a large number of sensors. *Mathematics of Control, Signals and Systems*, 1988.
10. J. N. Tsitsiklis. Decentralized detection. *Advances in Statistical Signal Processing*, 1993.
11. P. K. Varshney. *Distributed Detection and Data Fusion*. Springer Science & Business Media, 2012.
12. E. Wong and B. Hajek. *Stochastic Processes in Engineering Systems*. Springer Science & Business Media, 2012.
13. J. Hajnal and M. Bartlett. Weak ergodicity in non-homogeneous markov chains. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 54, pages 233–246. Cambridge Univ Press, 1958.
14. A. Jadbabaie, P. Molavi, and A. Tahbaz-Salehi. Information heterogeneity and the speed of learning in social networks. *Columbia Business School Research Paper*, (13-28), 2013.
15. A. Lalitha, A. Sarwate, and T. Javidi. Social learning and distributed hypothesis testing. In *Information Theory (ISIT), 2014 IEEE International Symposium on*, pages 551–555, June 2014. Extended version at arXiv 1410.4307.
16. N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1996.
17. P. Molavi, A. Tahbaz-Salehi and A. Jadbabaie. Foundations of Non-Bayesian Social Learning (August 2017). *Columbia Business School Research Paper No. 15-95*. Available at SSRN: <https://ssrn.com/abstract=2683607> or <http://dx.doi.org/10.2139/ssrn.2683607>, 2017.
18. A. Nedić, A. Olshevsky, and C. A. Uribe. Nonasymptotic convergence rates for cooperative learning over time-varying directed graphs. *arXiv preprint arXiv:1410.1977*, 2014.
19. K. R. Rad and A. Tahbaz-Salehi. Distributed parameter estimation in networks. In *IEEE Conference on Decision and Control (CDC)*, pages 5050–5055. IEEE, 2010.
20. S. Shahrampour and A. Jadbabaie. Exponentially fast parameter estimation in networks using distributed dual averaging. In *IEEE Conference on Decision and Control (CDC)*, pages 6196–6201. IEEE, 2013.
21. S. Shahrampour, A. Rakhlin, and A. Jadbabaie. Distributed detection: Finite-time analysis and impact of network topology. *arXiv preprint arXiv:1409.8606*, 2014.
22. S. Shahrampour, A. Rakhlin, and A. Jadbabaie. Finite-time analysis of the distributed detection problem. *arXiv preprint arXiv:1512.09311*, 2015.
23. J. Wolfowitz. Products of indecomposable, aperiodic, stochastic matrices. In *Proceedings of the American Mathematical Society*, pages 733–737. JSTOR, 1963.
24. N. H. Vaidya. Matrix Representation of Iterative Approximate Byzantine Consensus in Directed Graphs. arXiv 1203.1888, 2012.
25. D. Bajović, D. Jakovetić, J. M. Moura, J. Xavier, and B. Sinopoli. Large deviations performance of consensus+ innovations distributed detection with non-gaussian observations. *Signal Processing, IEEE Transactions on*, 60(11):5987–6002, 2012.
26. F. S. Cattivelli and A. H. Sayed. Distributed detection over adaptive networks using diffusion adaptation. *Signal Processing, IEEE Transactions on*, 59(5):1917–1932, 2011.
27. D. Jakovetić, J. M. Moura, and J. Xavier. Distributed detection over noisy networks: Large deviations analysis. *Signal Processing, IEEE Transactions on*, 60(8):4306–4320, 2012.
28. M. Pease & R. Shostak & L. Lamport. Reaching agreement in the presence of faults. *J. ACM* 27 (2), 228–234, Apr. 1980.
29. D. Dolev, N. A. Lynch, S. S. Pinter, E. W. Stark, and W. E. Weihl. Reaching approximate agreement in the presence of faults. *J. ACM*, 33(3):499–516, May 1986.
30. A. D. Fekete. Asymptotically optimal algorithms for approximate agreement. *Distributed Computing*, 4(1):9–29, 1990.
31. H. J. LeBlanc, H. Zhang, S. Sundaram, and X. Koutsoukos. Consensus of multi-agent networks in the presence of adversaries using only local information. In *Proceedings of the 1st International Conference on High Confidence Networked Systems*, HiCoNS ’12, pages 1–10, New York, NY, USA, 2012. ACM.
32. N. H. Vaidya. Iterative byzantine vector consensus in incomplete graphs. In *Distributed Computing and Networking*, pages 14–28. Springer, 2014.
33. N. H. Vaidya and V. K. Garg. Byzantine vector consensus in complete graphs. In *Proceedings of the 2013 ACM symposium on Principles of distributed computing*, pages 65–73. ACM, 2013.
34. N. H. Vaidya, L. Tseng, and G. Liang. Iterative approximate byzantine consensus in arbitrary directed graphs. In

- Proceedings of the 2012 ACM symposium on Principles of distributed computing*, pages 365–374. ACM, 2012.
35. H. Mendes and M. Herlihy. Multidimensional approximate agreement in byzantine asynchronous systems. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 391–400, New York, NY, USA, 2013. ACM.
  36. M.A. Perles and M. Sigron. A generalization of Tverberg's theorem. arXiv. 0710.4668, 2007.
  37. A. Nedić, A. Olshevsky, and C. A. Uribe. Network Independent Rates in Distributed Learning arXiv preprint arXiv:1509.08574, 2015.
  38. L. Su, and N. H. Vaidya. Asynchronous Distributed Hypothesis Testing in the Presence of Crash Failures University of Illinois at Urbana-Champaign, Tech. Rep, 2016. arXiv: 1606.03418.
  39. L. Su, and N. H. Vaidya. Non-Bayesian Learning in the Presence of Byzantine Agents. to appear in Proceedings of ACM Symposium on Distributed Computing (DISC), 2016

## A Proof of Equation (21)

First it is easy to see that

$$\begin{aligned} & \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \left| \log \frac{\ell_i(w_i | \theta_1)}{\ell_i(w_i | \theta_2)} \right| \\ & \geq \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \log \frac{\ell_i(w_i | \theta_1)}{\ell_i(w_i | \theta_2)} \end{aligned} \quad (30)$$

Let  $(\tilde{\theta}_1, \tilde{\theta}_2)$  and  $\tilde{w}_i$  be the hypotheses ordered pair and the private signal such that

$$\max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \left| \log \frac{\ell_i(w_i | \theta_1)}{\ell_i(w_i | \theta_2)} \right| = \left| \log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} \right|$$

If  $\log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} < 0$ , it holds that

$$\begin{aligned} \left| \log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} \right| &= -\log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} = \log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_2)}{\ell_i(\tilde{w}_i | \tilde{\theta}_1)} \\ &\leq \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \log \frac{\ell_i(w_i | \theta_1)}{\ell_i(w_i | \theta_2)}. \end{aligned} \quad (31)$$

If  $\log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} \geq 0$ , then

$$\begin{aligned} \left| \log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} \right| &= \log \frac{\ell_i(\tilde{w}_i | \tilde{\theta}_1)}{\ell_i(\tilde{w}_i | \tilde{\theta}_2)} \\ &\leq \max_{\theta_1, \theta_2 \in \Theta; \theta_1 \neq \theta_2} \max_{w_i \in \mathcal{S}_i} \log \frac{\ell_i(w_i | \theta_1)}{\ell_i(w_i | \theta_2)}. \end{aligned} \quad (32)$$

Equations (30), (31) and (32) together prove (21).