

# Gossip Average Consensus in a Byzantine Environment Using Stochastic Set-Valued Observers

Daniel Silvestre, Paulo Rosa, Rita Cunha, João P. Hespanha, Carlos Silvestre

**Abstract**—We address the problem of a consensus system in the presence of Byzantine faults seen as an attacker injecting a perturbation in the state of the nodes. We propose the use of Set-Valued Observers to detect if the state observations are compatible with the system dynamics. The method is extended to the stochastic case by introducing a strategy to construct a set that is guaranteed to contain all possible states with, at least, a pre-specified desired probability. The proposed algorithm is stable in the sense that it requires a finite number of vertices to represent polytopic sets while also enabling the a priori computation of the largest magnitude of a disturbance that an attacker can inject without being detected.

## I. INTRODUCTION

The problem of consensus relates to a set of agents agreeing on a common value using a distributed algorithm. In particular, we are interested in randomized gossip average consensus in that nodes are allowed to send messages to a random neighbor in order to compute the average of their initial state. Randomized gossip algorithms are designed to cope with “crash type” faults by using redundancy and randomization. However, Byzantine faults, such as an intruder in the system, can prevent convergence or drive the steady state of the system to any value [1]. Thus, the problem of detecting Byzantine faults is of prime importance in distributed algorithms and its applicability ranges from consensus algorithms to information dissemination and distributed control of industry processes.

The problem of detecting Byzantine faults using unreliable fault detectors was introduced in [2], where multiple classes of theoretic detectors are presented. For the specific case of a consensus system, an algorithm is proposed that makes use of an unreliable detector to solve the problem of consensus. However, [2] differs from our work in that the consensus value is assumed to be one of the initial values, whereas we are looking at asymptotically reaching the average of the initial values by using a linear dynamic system.

The consensus problem has been widely studied when considering rather non-antagonistic failure models which include packet drops and nodes leaving the network but, to enable a more comprehensive model, Byzantine faults

must be considered. The research interest in this issue has motivated a number of contributions including the scenario of unreliable networks in distributed systems. In particular, [1] considers the problem of detecting and correcting the state of the system in the presence of a Byzantine fault. The case of malicious agents and faulty agents is studied and the authors provide, in both cases, bounds on the number of corrupted nodes to ensure detectability of the fault. In [1], the system dynamics are described by a linear time-invariant model that constrains the communication in each time slot to be from a fixed set of senders to a set of receivers. Here, however, a randomized gossip algorithm is considered, thus dropping the assumption that the same sets of nodes are involved in message exchanges at every time instant.

The main contributions of this paper are as follows:

- the analysis of the problem of detecting an intruder in a randomized gossip consensus algorithm is recast into the framework of Linear Parameter-Varying (LPV) systems with uncertain dynamics;
  - an upper bound on the magnitude of the attacker signal is derived beyond which the attacker can be detected;
  - the concept of stochastic Set-Valued Observers (SVOs) is introduced by taking advantage of the use of  $\alpha$ -confidence sets, i.e., sets where the state of the system is known to belong with a desired pre-specified  $1 - \alpha$  probability; which can be viewed as a generalization of confidence intervals.
  - finally, it is also shown that this method inherits the main properties regarding the computational stability by having a bound on the volume and number of vertices needed to define the polytopic sets where the state is contained with a pre-specified probability; guarantees of intruder detection are also provided.
- on the number of vertices necessary to define the polytope sets and guarantees of intruder detection of the deterministic SVO-based model falsification approach.

Besides the introduction of a theoretical framework to address the problem, it is also needed to cover the mathematical machinery required to cope with the computation of the set where the current state can take values. From the random behavior of the gossip algorithm, for each possible transmission, the state can take values in a set of possible state realizations originated by that transmission and the previous state. To consider the worst case scenario, one needs to perform the union of all possible sets. Thus, at each transmission time, the algorithm must compute the set of possible states generated by each transmission and

D. Silvestre, R. Cunha, C. Silvestre are with the Dep. of Electrical and Computer Engineering, Instituto Superior Técnico, ISR, 1046-001 Lisboa, Portugal. {dsilvestre,rita,cjs}@isr.ist.utl.pt. This work is partially funded with grant SFRH//BD/71206/2010, from Fundação para a Ciência e a Tecnologia.

P. Rosa is with Deimos Engenharia, Lisbon, Portugal.

João P. Hespanha is with the Dept. of Electrical and Computer Eng., University of California, Santa Barbara, CA 93106-9560, USA. J. Hespanha was supported by the U.S. Army Research Laboratory and the U.S. Army Research Office under grants No. W911NF-09-1-0553 and W911NF-09-D-0001. hespanha@ece.ucsb.edu

compute the union of all of them. By definition, the number of sets grows exponentially with  $N$ . The concept of Set-Valued Observers (SVOs) was first introduced in [3] and [4] and further information can be found in [5] and [6] and the references therein.

The choice for representing the set of possible states depends on a mathematical formulation that enables fast and non-conservative intersections and unions of sets, as those are major and normally time-consuming operations when implemented in a computer. One alternative is to use the concept of zonotopes, described in [7] and further developed in [8] and [9]. However, it is normally the case that each proposal represents a compromise between the speed of the reunions and intersections. An alternative approach is adopted in this article, as described in the sequel, in order to attain the desired convergence guarantees, while keeping the computational requirements to a tractable level.

The applicability of the proposed method in the detection of faults in a randomized gossip algorithm is not limited to the consensus problem, as several challenges in the Fault Detection and Isolation (FDI) literature - [10], [11] - share the framework described in the sequel. In [12], the authors propose the use of SVOs for fault detection by using a model falsification approach. This paper extends the results in [12] to detect Byzantine faults in the consensus system, by rewriting its dynamics as an LPV. Moreover, unlike the approach in [12], the method proposed herein takes into account the information related to the probability of having a given communication.

The study of the FDI problems has been a long standing research topic, since the early 70's (see [13]), but still poses remarkable challenges both to the scientific community and the industry (see, for example, the survey in [14] and references therein). Classical fault detection methods such as the ones proposed in [13], [15], [16], [17], [18], [19] and [20], rely on designing filters that generate residuals that should be *large* under faulty environments. These strategies aim to derive bounds (or thresholds) on these residuals that can be used to decide whether a fault has occurred or not. However, calculation of these thresholds is typically cumbersome or poses stringent assumptions on the exogenous disturbances and measurement noise acting upon the system. The solution adopted in this paper alleviates the design complexity, while posing mild assumptions on the system. However, it also requires increased computational power when compared to classical FDI methodologies.

The remainder of this paper is organized as follows. In Section II, we describe the problem of randomized gossip consensus in the presence of Byzantine faults in Section II. The proposed solution is given in Section III and the main properties of the obtained results are stated in Section IV. Concluding remarks and potential future work are presented in Section VI.

*Notation* : The transpose of a matrix  $A$  is denoted by  $A^\top$ . For vectors  $a_i$ ,  $(a_1, \dots, a_n) := [a_1^\top \dots a_n^\top]^\top$ . We let  $\mathbf{1}_n := [1 \dots 1]^\top$  and  $\mathbf{0}_n := [0 \dots 0]^\top$  indicate  $n$ -dimensional vector of ones and zeros, respectively, and  $I_n$  denotes the

identity matrix of dimension  $n$ . Dimensions are omitted when clear from context. The vector  $e_i$  denotes the canonical vector whose components are equal to zero, except for the  $i$ th component. The notation  $\|\cdot\|$  refers to  $\|v\| := \sup_i |v_i|$  for a vector, and  $\|A\| := \bar{\sigma}(A)$ .

## II. PROBLEM STATEMENT

We consider a set of  $m$  agents, each of which with scalar state  $x_i(t)$ ,  $1 \leq i \leq m$  running a distributed iterative algorithm that guarantees convergence of the state to its initial average value, i.e.,

$$\lim_{t \rightarrow \infty} x_i(t) = x_{av} := \frac{1}{m} \sum_{i=1}^m x_i(0). \quad (1)$$

We refer to this problem as the *average consensus problem*.

In a gossip framework, at each transmission time, each node  $i$  chooses a random out-neighbor  $j$  according to the probability  $w_{ij}$ . Only nodes involved in the communication can change their state according to the received information. Thus, we regard transmission times as a discrete variable  $k$  that corresponds to the continuous variable  $k$  as between communication the state remains constant.

The communication topology is modeled by a graph  $G = (V, E)$ , where  $V$  represents the set of  $m$  agents, also denoted by nodes, and  $E \subseteq V \times V$  is the set of communication links. Node  $i$  can send a message to node  $j$ , if  $(i, j) \in E$ . If there exists at least one  $i \in V$  such that  $(i, i) \in E$  we say that the graph has self-loops, which can model, for example, packet drops, since node  $i$  only has access to its own value at that transmission time. We associate to graph  $G$  a *weighted adjacency matrix*  $W$  with entries:

$$W_{ij} := \begin{cases} w_{ij}, & \text{if } (i, j) \in E \\ 0, & \text{otherwise} \end{cases}, \quad (2)$$

where the weights  $w_{ij} \in [0, 1]$ .

We consider here randomized gossip algorithms in a Byzantine environment of the form

$$x(k+1) = U(k)x(k) + B(k)u(k), \quad (3)$$

where the matrix  $U(k)$  is selected randomly from a set  $\{Q_{ij}, (i, j) \in E\}$  and  $u(k)$  models Byzantine faults. The random selection of  $U(k)$  models the process by which nodes select a random out-neighbor, as described above. The input  $u(k)$  models the fact that some of the nodes may either report incorrect values regarding their state value or update their state by something other than the "fault-free" averaging rule

$$x(k+1) = U(k)x(k) \quad (4)$$

Matrices  $Q_{ij}$  implement the update on state variables  $x_i$  and  $x_j$  caused by a transmission from node  $i$  to node  $j$  and represent a set of *column stochastic* matrices (i.e.  $\mathbf{1}^\top Q_{ij} = \mathbf{1}^\top$ ) to keep the average between iterations.

In this paper, we assume symmetry in the communication and use as definition for the matrices  $U(k)$  the algorithm proposed in [21], which we recall here for readability. The matrices  $Q_{ij}$  that update the state are given by:

$$Q_{ij} = I - \frac{(e_i - e_j)(e_i - e_j)^T}{2}$$

A consensus system  $S$ , as defined above, refers to the pair of equations:

$$\begin{cases} x(k+1) = U(k)x(k) + B(k)u(k) \\ y(k) = C(k)x(k) \end{cases} \quad (5)$$

With a slight abuse of notation, we use  $y(k)$  to refer to the output of the system at time  $k$  and  $y_k(x(0), u_k)$  to express the same output as a function of the initial state  $x(0)$  and input  $u_k$ , where  $u_k$  denotes the vector of inputs up to time  $k$ .

The main goal of this paper can therefore be stated as: developing algorithms for detecting nonzero inputs  $u(k)$  in (3) that do not require knowledge of the matrices  $B(k)$  and signal  $u(k)$  in (3) and, instead, only use the measurements  $y_k$  which stands for all the measurements up to time  $k$ .

To formalize this problem, we introduce the following definition:

*Definition 1 (undetectable faults):* Take a consensus system  $S$  as in (5). A given nonzero input sequence  $u_k$  (corresponding to a fault) is said to be *undetectable in  $N$  iterations* if:

$$\forall k < N, \exists x(0), x'(0) \in \mathbb{R}^m : y_k(x(0), u_k) = y_k(x'(0), 0)$$

□

The intuition behind this definition is that a fault is only detectable if there is no possible set of initial conditions such that the sequence  $y(0) \cdots y(N)$  of measurable states can be generated without an attacker signal.

*Assumption 2 (detectable faults):* Each fault considered can be defined by means of an input sequence  $u_k$ , and is detectable in the sense of Definition 1.

The fault being detectable relates to the observability of the system (see [22]).

*Assumption 3 (bounded state):*  $\forall k < N, \|x(k)\| < c$  for a given constant  $c$ .

Assumption 3 is sustained by the fact that a non-faulty gossip algorithm has a bounded state. Therefore, there exists a constant  $c$  that if the state is larger one could trivially detect the occurrence of the fault.

### III. PROPOSED SOLUTION

We start by introducing the problem from a worst-case scenario perspective and by finding suitable computational machinery to detect Byzantine faults. Progress is then made to generalize the algorithm to include the probability information and convert the detection into a stochastic scenario.

#### A. Worst-case scenario

In a worst-case scenario, all the realization of  $U_k$  are assumed possible and the “low probability” of specific events cannot be used to infer the likelihood of a fault. Firstly, we

start by rewriting matrices  $U_k$  in (3) as a central matrix and a sum of uncertainties:

$$U_k = A_0 + \sum_{\ell=1}^{n_\Delta} \Delta_\ell(k) A_\ell \quad (6)$$

where  $n_\Delta$  is the number of required uncertainties for this representation, and each  $\Delta_\ell$  is a scalar uncertainty with  $|\Delta_\ell| = 1$ .

The dynamics of the system can now be cast into a Linear Parameter-Varying (LPV) model with uncertainty in the time-varying matrix  $A$ . Indeed, the dynamics in (3) can be rewritten as:

$$x(k+1) = \left( A_0 + \sum_{\ell=1}^n \Delta_\ell(k) A_\ell \right) x(k) + B_k u(k) \quad (7)$$

with matrix  $B_k$  having the column vector of ones and zeros according to whether or not a node presents a Byzantine fault behaviour. Detecting a fault in a worst-case scenario reduces to finding whether a given sequence of observations,  $y_k$ , can be generated by the dynamics in (7) with zero terms  $B_k u(k)$  or not, for any admissible initial conditions  $x(0)$ .

*Definition 4 (Distinguishability [23]):* Let  $S_A$  and  $S_B$  be two systems of the form (5) with outputs at discrete time  $k$ ,  $y_A(k)$  and  $y_B(k)$ , and initial states  $x_A(0)$  and  $x_B(0)$ , respectively. Then,  $S_A$  and  $S_B$  are said distinguishable in  $N \geq 0$  measurements if, for any

$$\begin{aligned} & (x_A(0), x_B(0), \phi(0), \dots, \phi(N-1)) \\ & \in \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^{n_\phi}, \dots, \mathbb{R}^{n_\phi} \end{aligned}$$

there exists  $k \in \{0, 1, \dots, N-1\}$  such that

$$y_A(k) \neq y_B(k).$$

where  $\phi$  is the vector of disturbances, uncertainties and inputs.

This notion of distinguishability in Definition 4 is closely related to the notion of fault detectability in Definition 1 introduced before. We study fault detectability by considering the distinguishability between the measurements of the real system and the measurements generated by a virtual “fault-free” model characterized by zero input.

We use the SVO framework from [23] and take advantage of the distinguishability concept to derive a bound on the magnitude of the injected attacker signal, such that the attack is detected whenever this bound is exceeded. This will be one of the main contributions of this paper.

It should be noticed that the distinguishability definition used for detecting a Byzantine fault is combinatorial by nature. Computing the set of possible state realizations is a matter of making the union of the possible state realizations for each combination of transmissions. In (6), that behavior is modeled by the vector of uncertainties  $\Delta_\ell(k)$ . Thus, the problem grows exponentially as the number of measurements  $N$  increases.

We adopt a similar notation as in [24] and define, at transmission time  $k$ ,  $X(k) := \text{Set}(M_k, m_k)$  where  $\text{Set}(M, m) :=$

$\{q : Mq \leq m\}$  represents a convex polytope containing the vectors  $q \in R^n$  that satisfy the constraint  $Mq \leq m$ , where  $\leq$  is a component-wise operation between the two vectors. The aim of an SVO is to find an approximation of the smallest set containing all possible states of the system, at time  $k$ ,  $\tilde{X}(k)$  with the knowledge that  $\forall 0 \leq i \leq N, x(k-i) \in \tilde{X}(k-i)$  and that the dynamics of the system are as in (7). Assumption 3 is needed to include the initial state into a polytope and use the SVO technique to compute a convex over-approximation. In other words, at each time  $k$ ,  $\tilde{X}(k)$  is an approximation of the set containing all possible states,  $X(k)$ , such that  $X(k) \subseteq \tilde{X}(k)$ .

More precisely, the initial state  $x(0) \in X(0)$  where  $X(0) := \text{Set}(M_0, m_0)$  and we can select  $M_0$  and  $m_0$  such that the corresponding polytope is guaranteed to contain the initial state. If  $\Delta$  is known, then the set  $X(k+1) := \text{Set}(M_{\Delta^*}(k+1), m_{\Delta^*}(k+1))$ , which contains all the possible states of the system at time  $k+1$ , can be found explicitly by

$$\underbrace{\begin{bmatrix} M(k)(A_0 + A_{\Delta^*})^{-1} \\ C(k+1) \\ -C(k+1) \end{bmatrix}}_{M(k+1)} x(k+1) \leq \underbrace{\begin{bmatrix} m(k) \\ y(k+1) \\ -y(k+1) \end{bmatrix}}_{m(k+1)} \quad (8)$$

where

$$A_{\Delta^*} = \sum_{\ell=1}^{n_{\Delta}} \Delta_{\ell}^* A_{\ell}$$

and  $\Delta_{\ell}^*$  is the realization of the uncertainty for the current transmission time. This procedure assumes an invertible transmission matrix. When this is not the case, we can adopt the strategy in [25] and solve the inequality

$$\begin{bmatrix} I & -A_0 - A_{\Delta^*}(k) \\ -I & A_0 + A_{\Delta^*}(k) \\ C(k+1) & 0 \\ -C(k+1) & 0 \\ 0 & M(k) \end{bmatrix} \begin{bmatrix} x(k+1) \\ x(k) \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ y(k+1) \\ -y(k+1) \\ m(k) \end{bmatrix} \quad (9)$$

by applying the Fourier-Motzkin elimination method [26] to remove the dependence on  $x(k)$  and obtain the set described by  $M(k+1)x(k+1) \leq m(k+1)$ .

Let the coordinates of each vertex of the hypercube  $H := \{\delta \in \mathbb{R}^N : |\delta| \leq 1\}$  be denoted by  $\theta_i, i = 1, \dots, 2^{n_{\Delta}}$ . Using (8) (or (9)) we compute  $X_{\theta_i}(k)$  with  $\Delta^* = \theta_i$ . Thus, the set of all possible states at time  $k+1$  can be obtained by

$$X(k+1) = \bigcup_{\theta_i \in H} \text{Set}(M_{\theta_i}, m_{\theta_i})$$

where we make the reunion for all the vertices  $\theta_i$  that appear in the graph, and where  $M_{\theta_i}$  and  $m_{\theta_i}$  are obtained using (8) or (9). It should be noticed that not all the vertices of the hypercube  $H$  are of interest, as some communications never take place due to the limited connectivity of the network. The convex hull,  $\tilde{X}(k+1)$ , of set  $X(k+1)$  is then obtained by using the methods described in [12], since, in general, the set  $X(k+1)$  is non-convex even if  $X(k)$  was convex. We recall Propositions 6.1 and 6.2 in [24] for completeness.

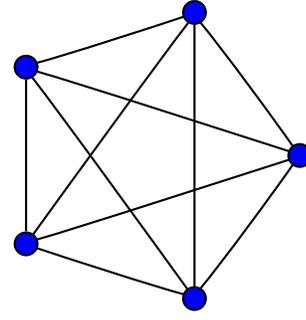


Fig. 1: 5-node complete graph.

*Proposition 5 (Membership of  $X(k)$  in  $\tilde{X}(k)$ ):* Consider a system described by (7) and assume that  $X(0) \subseteq \tilde{X}(0)$ . Then,  $X(k) \subseteq \tilde{X}(k)$  for all  $k \in \{0, 1, 2, \dots\}$ .  $\square$

*Proposition 6 (Growth of  $\tilde{X}(k)$ ):* Suppose that a system described by (7) with  $x(0) \in X(0)$  and  $u(k) = 0, \forall k$ , satisfies, for a sufficiently large  $N^*$ ,

$$\gamma_N := \max_{\substack{\Delta(k), \dots, \Delta(k+N) \\ |\Delta(m)| \leq 1, \forall m \\ k \geq 0}} \left\| \prod_{j=k}^{k+N} \mathcal{A}(j) \right\| < 1,$$

for all  $N \geq N^*$ , and where

$$\mathcal{A}(j) := \left[ \mathcal{A}_0(j) + \sum_{i=1}^{n_{\Delta}} A_i(j) \Delta_i(j) \right].$$

Then,  $\tilde{X}(k)$  cannot grow unboundedly.  $\square$

In summary, from Propositions 5 and 6, and a bound on the perturbation, one concludes that  $X(k) \subseteq \tilde{X}(k), \forall k$  if  $X(0) \subseteq \tilde{X}(0)$ . Moreover, the set  $\tilde{X}(t_k)$  cannot grow without bound, in the sense that the corresponding volume is bounded, and that there is a hyper-parallelepiped that, at each time, contains the set  $\tilde{X}(k)$ , and has a bounded distance between any two vertices.

Notice that using the method provided before to compute  $M(k)$  and  $m(k)$  for the “fault-free” model gives a set where our measurements can take values (Proposition 5 is essential to establish it). By doing the intersection with the vector of measurements  $y(k)$  for the real system, if it results in an empty set, the real system is distinguishable from the virtual “fault-free” model and a fault is detectable. In addition, in reference to Proposition 6, we can always derive a bounded set with a finite number of vertices to contain the set of actual possible states,  $X(k)$ .

### B. Stochastic Set-Valued Observers

The worst-case scenario considered in the previous section completely ignored the probabilistic structure behind the selection of the matrices  $U(k)$ .

To understand how this information can help in detecting Byzantine faults, consider the 5-node network ( $m = 5$ ) in Figure 1 and time horizon to detect the fault  $N = 20$ . Each node  $i$  takes a measurement  $x_i(0)$  of a quantity of

interest and then a consensus procedure starts, in order to calculate the average of the initial values of the nodes  $x_{av} = \sum_{i=1}^5 x_i(0)/5$ . Let us assume that the packet drop probability is known. In particular, let  $p_{drop} = 0.01$ , which means that the matrix of probabilities that guarantees the fastest convergence, calculated using a semi-definite program as in [21], is given by

$$W = \frac{(1 - p_{drop})}{m-1} \mathbf{1}\mathbf{1}^\top + \frac{mp_{drop} - 1}{m-1} I$$

where a packet drop is represented as a transmission from node  $i$  to itself, using the transmission matrix  $Q_{ii} = I$ . Each node is chosen with probability  $\frac{1}{m}$  and each matrix  $Q_{ij}$  representing a successful transmission from node  $i$  to  $j$  has probability  $\frac{W_{ij}}{m}$ .

If a node is not involved in a communication, it is only able to determine its own state. Suppose that the states of the agents start dissimilar from each other but that during the first  $N$  time steps, all agents are faulty and keep their states unchanged, i.e.,  $x(k) = x(0), \forall k \leq N$ . This fault is undetectable according to Definition 1 since there is a sequence of matrices  $U(k)$  that mimic the same behavior, which is a sequence of 20 failed transmissions due to the physical medium. Consequently, if the algorithm in the previous section is used,  $x(k) = x(0)$  must remain in the set  $\tilde{X}(k), \forall k$  and therefore a fault will not be detected. However, the probability of obtaining the sequence  $x(k) = x(0), \forall k \leq N$  is exceedingly small:

$$\text{Prob}\{x(k) = x(0), \forall 0 \leq k \leq 20\} = 10^{-40}$$

and is more likely to be a Byzantine fault. The inability of the SVO to incorporate the probability associated with each event is, therefore, a main drawback. Such an example motivates the introduction of Stochastic Set-Valued Observers (SSVOs) where the polytope containing the possible state is associated with a probability. The objective of this section concerns with extending the SVO concept to cope with the probability of getting a given sequence of measurements.

Consider the algorithm described in the previous subsection to generate the sets  $\tilde{X}(k)$ . The set for the SSVO  $\tilde{X}(k)$  is an  $\alpha$ -confidence set defined as:

$$\tilde{X}(k) := \alpha\text{-CS}\left(\bigcup_{\theta_i \in H} \text{Set}(M_{\theta_i}(k), m_{\theta_i}(k)), \alpha\right) \quad (10)$$

where  $\alpha$  is the desired confidence level. Recall that we rewrote each  $Q_{ij}$  as in (6), therefore, associating with each hypercube vertex  $\theta_i$  a transmission matrix  $Q_{ij}$  with correspondent probability  $w_{ij}$ . Take the map  $\psi : H \mapsto E$  which gives the correspondence between the vertices of the hypercube  $H$  and the edges set  $E$  and let us collect the minimum number of vertices  $\theta_i$  in  $\Theta$  such that  $\sum_{\theta_i} w_{\psi(\theta_i)} \geq 1 - \alpha$ . Then, we construct  $\tilde{X}(k)$ :

$$\tilde{X}(k) := \bigcup_{\theta_i \in \Theta} \text{Set}(M_{\theta_i}(k), m_{\theta_i}(k)) \quad (11)$$

Computationally, it requires to sort the vertices  $\theta_i$  according to probabilities  $w_{\psi(\theta_i)}$  as to construct the minimum  $\Theta$  and then determining  $M_{\theta_i}(k)$  and  $m_{\theta_i}(k)$  as before.

#### IV. MAIN PROPERTIES

In this section, we start by providing a result showing that the set generated by the SSVO is a  $\alpha$ -confidence set.

*Proposition 7:* Take the definition of  $\tilde{X}(k)$  as in (11). Then,  $\forall k, \tilde{X}(k)$  is a  $\alpha$ -confidence set.

*Proof:* The result is straightforward from the fact

$$\text{Prob}\left[x(k) \in \bigcup_{\theta_i \in \Theta} \text{Set}(M_{\theta_i}(k), m_{\theta_i}(k))\right] \geq \sum_{\theta_i \in \Theta} w_{\psi(\theta_i)} \geq 1 - \alpha$$

In view of Proposition 7, if we have a detectable fault as in Definition 1, and declare the fault to be detected when the set  $\tilde{X}(k)$  becomes empty, then, the probability of having a false positive is less or equal to  $\alpha$ . In addition, by the construction of the set  $\tilde{X}(k)$ , we observe that  $\tilde{X}(k)$  is  $\alpha$ -confidence set with  $\alpha = 0$  and we have  $\tilde{X}(k) \subseteq X(k)$ .

Let us borrow the definitions as in [24]:

$$(A_N, b_N) = LFM\left(\begin{bmatrix} M_N \\ -M_N \\ \tilde{M}_0 \\ \tilde{M}_W \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ \tilde{m}_0 \\ \tilde{m}_W \end{bmatrix}, 2m\right) \quad (12)$$

where the *LFM* stands for the left Fourier-Motzkin elimination method and:

$$\tilde{M}_0 = [\text{diag}(M_0, M_0) \quad 0 \quad 0 \quad 0], \quad \tilde{m}_0 = \begin{bmatrix} m_0 \\ m_0 \end{bmatrix},$$

$$\tilde{M}_W = [0 \quad \text{diag}(M_d, \dots, M_d)],$$

$$\tilde{m}_W = [m_d^\top \quad \dots \quad m_d^\top],$$

$$M_N = \left[ \begin{array}{cc|c} C_A & -C_B & \bar{R} \\ C_A A_A & -C_B A_B & \\ \vdots & \vdots & \\ C_A A_A^N & -C_B A_B^N & \end{array} \right],$$

$$\bar{R} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ R_1^1 & 0 & \dots & 0 \\ R_1^2 & R_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ R_1^N & R_2^N & \dots & R_N^N \end{bmatrix},$$

$$R_i^k = [C_A A_A^{k-i} B_A \quad -C_B A_B^{k-i} B_B].$$

where the  $M_d$  and  $m_d$  define the set for the signal  $u$ , i.e.,  $M_d$  and  $m_d$  are defined such that  $u(k) \in \text{Set}(M_d, m_d)$ ; and  $A_i, B_i$  and  $C_i$  with  $i \in \{A, B\}$  are the matrices from the two systems. With a slight abuse of notation, we are writing the product of  $N$  matrices  $A(k)$  as  $A^N$  for shorter notation.

We introduce the next proposition, which provides a theoretical bound on the magnitude of the attacker's signal for the fault to be detectable.

*Proposition 8 (Attacker signal bound [24]):* Let us consider a "fault-free" system:

$$S_A = \begin{cases} x_A(k+1) = A(k)x_A(k) \\ y_A(k) = C(k)x_A(k) \end{cases}$$

and a faulty system (5):

$$S_B = \begin{cases} x_B(k+1) = A(k)x_B(k) + B(k)u(k) \\ y_B(k) = C(k)x_B(k) \end{cases}$$

where  $u \in \mathbb{R}^{n_u}$ ,  $x_i \in \mathbb{R}^m$ ,  $y_i \in \mathbb{R}^2$ , initialized with the same initial conditions and let the  $X(k) = \text{Set}(A_N, b_N)$ , where the current state is contained, be defined as in (12).

Further define:

$$P_A = \frac{1}{N} \text{diag}(I_{n_u}, 0_{n_u}, \dots, I_{n_u}, 0_{n_u})$$

and

$$P_B = \frac{1}{N} \text{diag}(0_{n_u}, I_{n_u}, \dots, 0_{n_u}, I_{n_u})$$

and let  $\gamma_{min} \geq 0$  be defined such that

$$\gamma_{min} \geq \max_{A_N \xi \leq b_N} \xi^T P_A \xi, \text{ and } \gamma_{min} \geq \max_{A_N \xi \leq b_N} \xi^T P_B \xi$$

where the vector  $\xi$  stacks all the measurements, initial states and perturbation from the attacker. Then, system  $S_1$  and  $S_2$  are distinguishable in  $N$  measurements if

$$\frac{1}{N} \sum_{k=0}^N \|u(k)\|^2 > \gamma_{min} \quad (13)$$

□

Notice that in Proposition 8, the matrices  $A_N$  and the vector  $b_N$  can be constructed using and concatenating the successive  $M(k)$  and  $m(k)$  in (9). Also, the parameter  $\gamma_{min}$  is the smallest "disturbance" that an attacker can inject in the system before system  $S_1$  and  $S_2$  are distinguishable in the sense that the measured output of the faulty system cannot be generated by the dynamics of the non-faulty one. As a consequence, we can use the same line-of-thought to derive the following result.

*Corollary 9 (Attacker signal bound for SSVO):* Consider a non-faulty system  $S_1$  and a faulty system  $S_2$  as in Proposition 8. Then, the Byzantine fault is detectable in  $N$  measurements with a false positive probability lower or equal than  $\alpha$  if

$$\frac{1}{N} \sum_{k=0}^N \|u(k)\|^2 > \gamma_{min} \quad (14)$$

□

## V. SIMULATION RESULTS

In this section, we present simulation results with some meaningful scenarios that illustrate specific features of the proposed fault detection scheme. Two different types of faults are detected by the standard SVO and a third type is detected by the SSVO to motivate the use of the stochastic information, where a worst-case detection is not suitable.

The network used for our simulations is different from the one in Figure 1 as to make the detection more challenging. The intuition is that the fewer neighbours the detecting node has, the harder the detection is. If a node does not communicate with the detecting node, its state is not observable from the detecting node viewpoint. Without loss of generality, we illustrate the results from the perspective of one of the neighbors of a Byzantine node, i.e., the output  $y(k)$  corresponds to the observations of one of the neighbors of the Byzantine node.

We consider a 5-node network with nodes labelled  $i, i \in \{1, 2, 3, 4, 5\}$  and initial state  $x_i(0) = i - 1$  and a nominal bound for the state magnitude of  $|x_i| \leq 5$ . In order to reduce complexity and to study the properties of the algorithms in a disadvantageous setting, we considered  $N = 1$ , meaning that we only use the information from the previous iteration for the estimates. This is a worst-case scenario, as the algorithm only takes into account the dynamics of the system with one time step from the last estimate and discards prior observations and their propagation using multiple steps with the system dynamics. A missed detection is considered if the algorithm is not able to detect the fault within 300 measurements. Each result presented corresponds to 1000 Monte-Carlo runs. For convenience, node 1 is the node that performs the detection and node 2 is the failing node, and no faults occur in the first 10 transmissions. Note that if a node introduces Byzantine faults from the start of the algorithm, it can do so without being detected since the network has no information about the initial state of the Byzantine node. The following probability matrix is used:

$$W = \begin{bmatrix} 0 & 0.5 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.25 & 0 & 0.25 \\ 0.5 & 0.25 & 0 & 0.125 & 0.125 \\ 0 & 0 & 0.125 & 0.25 & 0.625 \\ 0 & 0.25 & 0.125 & 0.625 & 0 \end{bmatrix}$$

The first scenario corresponds to an erratic node failure in which the node will respond with a random value. Specifically, after 10 iterations the node replies as if its state was drawn uniformly from the interval of admissible states  $[-5, 5]$ .

Figure 2 depicts the histogram of the detection times for the aforementioned fault. In this simulation, the detection rate was 100%, which is not surprising from the erratic behaviour of the node. Analysing the distribution, one key observation that is recurrent in other simulations is that as times passes the detection is more provable. At the moment of detection, we have  $\gamma_{min} = 56.25$  and the correspondent magnitude of the injected signal  $\|u\|^2 = 4.405$ . We concluded that the

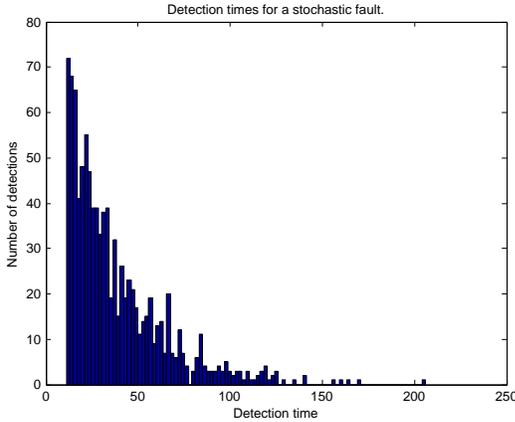


Fig. 2: Detection times for the stochastic fault.

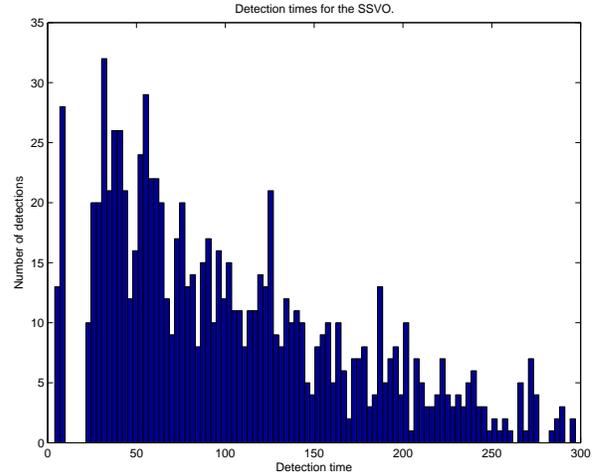


Fig. 4: Detection times for the SSVO.

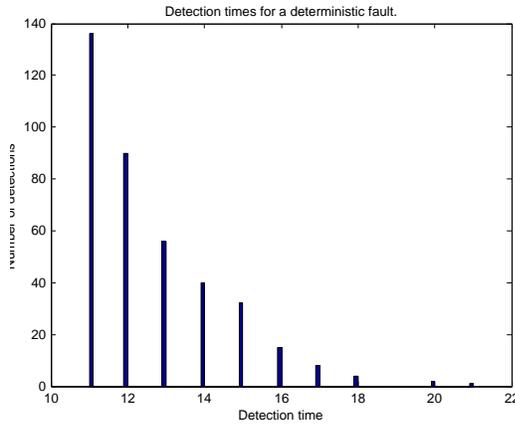


Fig. 3: Detection times for the deterministic fault.

value of  $\gamma_{min}$  as a worst-case scenario is conservative in the sense that signals with a smaller energy are also detected.

We also considered a less erratic scenario where a node becomes unresponsive due to CPU load or software crash, does not perform the consensus update and, therefore, replies always with the same value.

Figure 3 depicts the detection time for the deterministic fault where the node replies with the same value. In this case, the detection rate is 38.4%. In some sense, the lower detection rate is motivated by the fact that this fault does not change the state as much as the previous one. Since node 2 has other neighbors not in common with node 1, the fault is undetectable in more transmission sequences than in the previous simulation. Nonetheless, we still observe the behaviour that the fault is more likely to be detected as time progresses. Once again, we calculate  $\gamma_{min} = 76.56$  and  $\|u\|^2 = 2.997$  and observe that the injected signal is still detected even though its energy is less than the theoretical bound.

To illustrate the advantage of the SSVO when detecting faults, we consider a scenario where a node takes advantage of the network and initiates communication with a neighbour

regardless of the probability matrix  $P$  but does not change any of the nodes state. Notice that using an SVO, such faults would not be detected as any communication pattern that is possible is considered regardless of its probability. Between transmission time  $10 < k < 20$  the communication takes place between node 3 and 4 and  $\alpha = 0.1$ .

Figure 4 depicts the detection times for the SSVO case with a detection rate of 92.8%. Even though the behaviour is still the same, we can no longer guarantee that the detection is caused by the fault and not by a communication pattern which we consider to be a fault, but that has non-zero probability of happening without a fault.

## VI. CONCLUSIONS AND FUTURE WORK

The scenario of a consensus system subject to Byzantine faults was addressed, which is a larger group of faults than “crash” type failures modeled by a change in the communication graph (removing all the edges between nodes that are unable to communicate). This extension allows for faults such as intermittent failures, state corruption, nodes executing erroneous steps of the algorithm or corrupted messages that pass the medium checksums and other detection mechanisms, among many other possibilities. We are particularly interested in faults that can change the final consensus value (i.e., the steady state of the system) and on the magnitude of such a shift.

The paper focuses on gossip algorithms with stochastic transmissions that are robust to most of the crash type faults. We model the transmission as uncertainties in the system dynamics and present proofs of detection if the fault is *detectable* in a given sense, and while providing bounds on the magnitude of the attacker’s injected signal.

The proposed approach adopts the concept of set-valued observers to generate, at each transmission time, a convex set containing all possible state realizations and converting the problem to that of detecting whether the intersection between this set and the set of observations is void or not

(i.e. rewriting the problem as a distinguishability problem rather than a detectability one). We also provide an algorithm to construct confidence sets that take into account the probabilistic nature of the stochastic gossip algorithm, which reduces the conservativeness of the worst-case approach, at the expense of introducing a probability of false positives.

The set generated by an SVO is always the worst-case scenario and, for that reason, a superset of the set generated by an SSVO in the stochastic sense. Thus, all the properties of the SVOs in terms of bounded volume and number of vertices can be extended to the stochastic framework.

Our main contribution is two-folded: we introduce a novel type of observer for discrete-time stochastic systems, showing that the estimated set containing the state, at each time instant, is bounded; and a bound on the magnitude of the attacker's input signal above which the attack can be detected is derived.

We envisage some directions of future work by considering asymmetric communications and a broader class of distributed algorithms. Some additional work is necessary in reducing the computational complexity of calculating the set where the state of the system is. Finally, different types of faults may be analyzed in order to pursue less conservative conditions that can still guarantee the detectability of faults.

## REFERENCES

- [1] F. Pasqualetti, A. Bicchi, and F. Bullo, "Consensus computation in unreliable networks: A system theoretic approach," *Automatic Control, IEEE Transactions on*, vol. 57, no. 1, pp. 90–104, jan. 2012.
- [2] K. P. Kihlstrom, L. E. Moser, and P. M. Melliar-Smith, "Solving consensus in a byzantine environment using an unreliable fault detector," in *Proceedings of the International Conference on Principles of Distributed Systems (OPODIS)*, 1997, pp. 61–75.
- [3] H. Witsenhausen, "Sets of possible states of linear systems given perturbed observations," *Automatic Control, IEEE Transactions on*, vol. 13, no. 5, pp. 556–558, oct 1968.
- [4] F. Schweppe, "Recursive state estimation: Unknown but bounded errors and system inputs," *Automatic Control, IEEE Transactions on*, vol. 13, no. 1, pp. 22–28, feb 1968.
- [5] F. Schweppe., *Uncertain Dynamic Systems*. Prentice-Hall, 1973.
- [6] M. Milanese and A. Vicino, "Optimal estimation theory for dynamic systems with set membership uncertainty: An overview," *Automatica*, vol. 27, no. 6, pp. 997–1009, 1991.
- [7] D. Bertsekas and I. Rhodes, "Recursive state estimation for a set-membership description of uncertainty," *Automatic Control, IEEE Transactions on*, vol. 16, no. 2, pp. 117–128, apr 1971.
- [8] C. Combastel, "A state bounding observer for uncertain non-linear continuous-time systems based on zonotopes," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, dec. 2005, pp. 7228–7234.
- [9] T. Alamo, J. Bravo, and E. Camacho, "Guaranteed state estimation by zonotopes," *Automatica*, vol. 41, no. 6, pp. 1035–1043, 2005.
- [10] R. J. Patton, "Fault-tolerant control systems: The 1997 situation," in *IFAC symposium on fault detection supervision and safety for technical processes*, vol. 3, 1997.
- [11] J. Bokor and Z. Szabó, "Fault detection and isolation in nonlinear systems," in *Annual Reviews in Control* 33.2, 2009, pp. 113–123.
- [12] P. Rosa, C. Silvestre, J. Shamma, and M. Athans, "Fault detection and isolation of LTV systems using set-valued observers," in *Proceedings of the 49th IEEE Conference on Decision and Control*, December 2010, pp. 768–773.
- [13] A. S. Willmsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, no. 6, pp. 601–611, 1976.
- [14] I. Hwang, S. Kim, Y. Kim, and C. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *Control Systems Technology, IEEE Transactions on*, vol. 18, no. 3, pp. 636–653, may 2010.
- [15] J. D. Barrett, "Diagnosis and fault-tolerant control," *Technometrics*, vol. 49, no. 4, pp. 493–494, 2007.
- [16] J. Bokor and G. Balas, "Detection filter design for LPV systems – a geometric approach," *Automatica*, vol. 40, pp. 511–518, 2004.
- [17] G. Ducard, *Fault-tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles*, ser. Advances in industrial control. Springer, 2009.
- [18] A. Marcos, S. Ganguli, and G. J. Balas, "An application of h fault detection and isolation to a transport aircraft," in *Control Engineering Practice*, vol. 13, no. 1. Control Engineering Practice, 2005, pp. 105–119.
- [19] X. Ding and P. M. Frank, "Fault detection via factorization approach," *Systems and Control Letters*, vol. 14, no. 5, pp. 431–436, 1990.
- [20] S. Narasimhan, P. Vachhani, and R. Rengaswamy, "New nonlinear residual feedback observer for fault diagnosis in nonlinear systems," *Automatica*, vol. 44, pp. 2222–2229, 2008.
- [21] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.
- [22] M. Grewal and K. Glover, "Identifiability of linear and nonlinear dynamical systems," *IEEE Trans. on Automatic Control*, vol. 21, no. 6, pp. 833–837, 1976.
- [23] P. Rosa and C. Silvestre, "On the distinguishability of discrete linear time-invariant dynamic systems," in *Proceedings of the 50th IEEE Conference on Decision and Control*, December 2011.
- [24] P. Rosa, "Multiple-model adaptive control Multiple-Model Adaptive Control of Uncertain LPV Systems," Ph.D. dissertation, Technical University of Lisbon, 2011.
- [25] J. Shamma and K.-Y. Tu, "Set-valued observers and optimal disturbance rejection," *Automatic Control, IEEE Transactions on*, vol. 44, no. 2, pp. 253–264, feb 1999.
- [26] S. Keerthi and E. Gilbert, "Computation of minimum-time feedback control laws for discrete-time systems with state-control constraints," *Automatic Control, IEEE Transactions on*, vol. 32, no. 5, pp. 432–435, may 1987.