A Distributed Control Approach to Consensus within Byzantine Infrastructure

Massimo Albarello
malbarello@student.ethz.ch

Leonardo Barcotto
lbarcotto@student.ethz.ch

Andrea Da Col
adacol@student.ethz.ch

Alberto Loro
alloro@student.ethz.ch

***Abstract*—In this paper, inspired by a distributed ledger technology called IOTA [1], we model and analyze a simplified version of the IOTA consensus algorithm from a distributed control system point of view. First we propose a randomized distributed averaging algorithm and show that all nodes converge to consensus almost surely. Then, in order to make it robust to malicious nodes, we implement a control strategy on some nodes called coordinators that update their opinions to steer the collective behaviour of the network towards a given reference, thus achieving some kind of robustness against the influence of the malicious nodes themselves. We provide mathematical guarantees on some scenarios and conjectures on others based on simulations.**

I. Introduction

IOTA is a distributed ledger that can be used to exchange value between people and machines. All transactions are stored in all the nodes of the network in order to have an immutable but constantly updated record of all the past transactions. This allows each node to independently verify whether each new transaction is valid or not, depending on the past history of the ledger.

A distributed consensus algorithm is what allows all nodes in the network to agree on the current state of the ledger and is also what handles conflicts between transactions. When a new transaction A is signed, it is sent to an IOTA node which broadcasts it throughout the network until either it reaches all the nodes or a node detects another transaction B that is in conflict with the first one. Notice that in the first case there is no doubt whether the transaction A is legitimate or not. Our main focus is to analyze the conflict scenario and understand the conditions under which consensus on the valid transaction is achieved with an appropriate control strategy.

The two transactions A and B are in conflict when they try to spend the very same amount of money twice; this is called a *doublespend*. This situation might be encountered when a user broadcasts transaction A to one node and transaction B to another. Initially, both nodes will think that the received transaction is valid because they only received one of the two. However, as the two transactions propagate throughout the network, at some point a node will receive both of them and notice the doublespend. This is when the Fast Probabilistic Consensus (FPC) algorithm, described in [2] and empirically analyzed in [3], comes into play. The algorithm has to make sure that all nodes in the network reach consensus on one of the two transactions. It is important to notice that there is no ”wrong” or ”right” transaction; as long as both of them are signed by a private key of a wallet containing enough money, either of them can be the valid transaction. However, the network has to agree on which one of the two to choose as the only valid one.

In order to do this, IOTA implements the FPC algorithm. At the beginning of the algorithm every node has an initial binary opinion which is then updated in the following iterative way: at every step each node randomly queries p other nodes, averages their opinion with its own and sets this result as its opinion for the next round until some local termination rule is satisfied. Moreover the FPC algorithm provides some guarantees in the scenario where part of the nodes are malicious or byzantine, whose goal is to delay the consensus or prevent it from happening.

Inspired by this we decided to analyze, from a distributed control perspective, the conflict scenario and understand under which conditions the consensus on the valid transaction is achieved with an appropriate control strategy. Furthermore we investigate the scenario where a malicious node, modelled as a stubborn agent, is present. It is worth to mention that to use the distributed control techniques, we had to take a quite different approach that led us to make some simplifying assumptions.

The remainder of this paper is organized as follows. Section II provides a description of the system's model and the assumptions that have been made. Section III gives a mathematical analysis that employs the theoretical results developed during the lectures. Section IV contains the numerical simulations of the model as well as some considerations on the system's behaviour under different circumstances. Final considerations are contained in Section V.

II. Problem Setup

An underlying network of n agents with associated unweighted undirected graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ is considered. On top of this we define a sequence of weighted digraphs $\{\mathcal{G}_k(\mathcal{V},\mathcal{E}_k,\{a_e\}_{e\in\mathcal{E}_k})\}_{k\in\mathbb{Z}\geq 0}$, where $\mathcal{E}_k \subseteq \mathcal{E}$ for all $k \in \mathbb{Z}^+$. Note that each $\mathcal{G}_k$ is a spanning subgraph of $\mathcal{G}$.

Let $(\mathcal{A},\mathcal{F},\mathbb{P})$ be a probability space, with $\mathcal{A} = S_n = \{$set of row-stochastic matrices of order n with strictly positive diagonal entries$\}$, $\mathcal{F}$ the Borel σ-algebra of $\mathcal{A}$ and $\mathbb{P}$ a probability measure defined on $\mathcal{A}$. Since each $A \in \mathcal{A}$ can be associated to a digraph whose adjacency matrix is A itself, then the sequence $\{A(k)\}_{k\in\mathbb{Z}_{\geq 0}}$ of i.i.d. matrices uniquely defines a sequence of random digraphs $\{\mathcal{G}_k(\mathcal{V},\mathcal{E}_k,\{a_e\}_{e\in\mathcal{E}_k})\}_{k\in\mathbb{Z}\geq 0}$.

This definition is what we use to model the sampling process, where at time step k each node i samples p random nodes from $\mathcal{V}\setminus\{i\}$. Precisely, the sampling process consists in randomly drawing p edges $(i, j) \in \mathcal{E}$ such that $i \neq j$ $\forall i \in \{1, ..., n\}$ without replacement. In addition, every i always samples itself, i.e. every node has a self-loop at each round. This implies that each row of every $A \in \mathcal{A}$ has exactly $p + 1$ strictly positive entries or, equivalently, that the out-degree of each node in the generated digraph is constant at each time-step.

To describe the averaging of the opinions received with the random queries, we consider a system modelled by the following stochastic, discrete-time, linear dynamics:

$$x(k+1) = A(k)x(k) \tag{1}$$

where $x(k) \in [0, 1]^n$ is the vector of network opinions and the $A(k)$'s are independent, identically distributed (i.i.d) random matrices drawn from $\mathcal{A}$.
For the sake of clarity, the A's belonging to the set of possible events $\mathcal{A}$, which is in our case discrete and finite, have entries:

$$(A)_{ij} = \frac{1}{p+1} \begin{cases} 1, & \text{if } i \text{ samples } j \\ 0, & \text{if } i \text{ does not sample } j \end{cases}$$

where $A\mathbf{1}_n = \mathbf{1}_n$ and $A_{ii} = \frac{1}{p+1}$. In other words all A's are row-stochastic and with strictly positive diagonal entries.
What is important to understand is that the topology of the network at a certain time instance depends on a random graph process and it can change at every k, i.e. a sequence of random digraphs $\{\mathcal{G}_k\}_{k \in \mathbb{Z}_{\geq 0}}$ is obtained.
Since working with $\{A(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ is quite cumbersome, from now on we will often refer to the expectation of the averaging matrix $\mathbb{E}[A(k)]$ as our adjacency matrix. We use the fact that $\mathbb{E}[A(k)]$ is a good way to measure the evolution of the system, as it will be shown in Section IV. Recalling that:

$$\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$$

with X and Y independent random variables, and that the $A(k)$'s are independent and identically distributed random matrices we get, for all non-negative k's:

$$\begin{aligned} \mathbb{E}[A(k) \cdots A(0)] &= \mathbb{E}[A(k)] \cdots \mathbb{E}[A(0)] \\ &= \mathbb{E}[A(k)]^{k+1} \end{aligned}$$

Thus yielding the following simpler time-invariant dynamics:

$$x(k+1) = \mathbb{E}[A(k)]x(k) \tag{2}$$

Indeed, much information can be learned simply from the spectrum of $\mathbb{E}[A(k)]$, making randomized averaging algorithms easy to study.

We will work with two different assumptions on the underlying network $\mathcal{G}(\mathcal{V}, \mathcal{E})$: first, we assume $\mathcal{G}$ complete, and later we will weaken this assumption supposing $\mathcal{G}$ connected, with every node having a self-loop and out-degree at least $p + 1$. From now on we will refer to this second network topology as $\mathcal{G}^*$. Notice that in the latter case a node i will sample all the $j \neq i$ from $\mathcal{N}_i \setminus \{i\}$, with $\mathcal{N}_i$ the set of all the out-neighbours of i, based on the architecture of $\mathcal{G}^*$. Clearly $\mathcal{N}_i \setminus \{i\} \subseteq \mathcal{V} \setminus \{i\}$ in the general case, with the equality holding only in the case of complete graph.
Since all the neighbours can be sampled equally likely (in our case: $\frac{p}{n-1}$), it is straightforward to show that in the case where $\mathcal{G}$ is complete we have:

$$\Big(\mathbb{E}[A(k)]\Big)_{ij} = \frac{1}{p+1} \begin{cases} 1, & \text{if } i = j \\ \frac{p}{n-1}, & \text{if } i \neq j \end{cases} \tag{3}$$

while in the other case ($\mathcal{G}^*$):

$$\Big(\mathbb{E}[A(k)]\Big)_{ij} = \frac{1}{p+1} \begin{cases} 1, & \text{if } i = j \\ \frac{p}{n_i - 1}, & \text{if } j \in \mathcal{N}_i \setminus \{i\} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $n_i = \mid \mathcal{N}_i \mid$. Note that, even if $\mathcal{G}^*$ is undirected and thus its associated adjacency matrix is symmetric, $\mathbb{E}[A(k)]$ will not be symmetric in general.

In addition to this more general setup, which is not robust to any adversarial attack, the network will be later modelled to have n_s *standard agents*, n_c *coordinators* and n_b *byzantine agents*. Specifically, a standard agent will apply the averaging algorithm we have described, while a coordinator will modulate its opinion in order to make the state of the network track a given reference signal by means of a feedback control law. Instead, malicious agents are modelled to be stubborn, i.e. they don't update their initial state, but can still be queried by the standard agents and thus influence their opinion. This case will give rise, in general, to digraphs associated to the following upper block-triangular adjacency matrix:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ \mathbf{0}_{n_b \times (n_c + n_s)} & I_{n_b \times n_b} \end{bmatrix} \tag{5}$$

where $A_{11} \in \mathbb{R}^{(n_c+n_s) \times (n_c+n_s)}$ and $A_{12} \in \mathbb{R}^{(n_c+n_s) \times n_b}$. It is clear from (5) that malicious agents only sample themselves. Notice that A is row-stochastic and that the condensation digraph associated to $\mathbb{E}[A(k)]$ has n_b aperiodic sinks.
We will also assume the coordinators to have limited knowledge of the global state of the network, i.e. they can only directly measure the opinion of their neighbours. Therefore we will refer to the coordinators as having limited network visibility [4]. Two control strategies on the coordinators will be discussed, namely a proportional (P) and a proportional-integral (PI) feedback control law. The open-loop system under the P strategy is described by:

$$\begin{aligned} x(k+1) &= \mathbb{E}[A(k)]x(k) + Bu(k) \\ u(k) &= K_P(r - y(k)) \\ y(k) &= Cx(k) \end{aligned} \tag{6}$$

The augmented-state closed-loop system employing a PI controller is instead:

$$\begin{aligned} x(k+1) &= \mathbb{E}[A(k)]x(k) + Bu(k) \\ z(k+1) &= z(k) + (r - y(k)) \\ u(k) &= K_P(r - y(k)) + K_I z(k) \\ y(k) &= Cx(k) \end{aligned} \tag{7}$$

where $y(k)$, $u(k) \in \mathbb{R}^{n_c}$ are the measurement and control input signal vectors associated to the coordinators. B indicates which nodes are the coordinators. Without loss of generality we assume that the coordinators are positioned in the first n_c nodes, thus yielding:

$$B = \begin{bmatrix} I_{n_c \times n_c} \\ \mathbf{0}_{n_s \times n_c} \end{bmatrix}$$

C is a row-stochastic matrix where the i-th row contains the weights that specify a convex combination of opinions measured by the i-th coordinator:

$$C = \begin{bmatrix} c_{11} & \dots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n_c 1} & \dots & c_{n_c n} \end{bmatrix}$$

Specifically, in our analysis a measurement is the arithmetic mean computed by a coordinator on its neighbouring nodes' opinion. Hence the following distinction based on the network architecture is needed: if the graph is complete then all the coordinators will have global visibility of the network, namely they can directly measure the opinion of every other node. In the matrix formulation this means $c_{ij} = \frac{1}{n}$ for all i, j. When $\mathcal{G} = \mathcal{G}^*$, the coordinators necessarily have a myopic visibility of the network. Hence $c_{ij} = 0$ for some $\{i, j\}$ and the nonzero elements are of the form $c_{ij} = \frac{1}{n_i}$, where we remind the reader that n_i is potentially different for different coordinators, representing the fact that they control a different number of standard nodes.

For simplicity, we will assume the coordinators to have common constant reference signal $r \in \text{span}(\mathbf{1}_{n_c})$, and share both the proportional gain k_P and the integral gain k_I. Hence $K_P, K_I \in \mathbb{R}^{n_c \text{x} n_c}$ are diagonal matrices with constant diagonal entries k_P and k_I respectively.

Note that (2) is a particular case of both (6) and (7) with $B = \mathbf{0}_{n \times n_c}$.

III. Analysis

First of all, we want to investigate if the simple randomized system (1) is able to reach consensus.

Theoretical results on consensus for randomized algorithms are discussed in [6], where necessary and sufficient conditions are presented. The following theorem shows that consensus is indeed reachable given some assumptions.

Theorem 1 (Consensus for randomized algorithms)**.** *Let $\{A(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ be a sequence of random row-stochastic matrices with associated digraphs $\{G(k)\}_{k \in \mathbb{Z}_{\geq 0}}$. Assume:*

(A1) *the sequence of variables $\{A(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ is i.i.d.,*

(A2) *at each time k, the random matrix $A(k)$ has a strictly positive diagonal so that each digraph in the sequence $\{G(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ has a self-loop at each node, and*

(A3) *the digraph associated to the expected value matrix $\mathbb{E}[A(k)]$, for any k, has a globally reachable node.*

Then the following statements hold almost surely:

(i) *there exists a random non-negative vector $w \in \mathbb{R}^n$ with $w_1 + w_2 + ... + w_n = 1$ such that:*

$$\lim_{k \to \infty} A(k)A(k-1) \cdots A(0) = \mathbf{1} w^T$$

(ii) *as $k \to \infty$, each solution of $x(k{+}1) = A(k)x(k)$ satisfies*

$$\lim_{k \to \infty} x(k) = (w^T x(0)) \mathbf{1}_n.$$

It is easy to check that all the assumptions are satisfied for our model. (A1) and (A2) come directly from the problem setup. Recall that at time k every node queries independently of the previous time steps, hence the sequence $\{A(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ is made up of independent matrices. The sampling process is the same at every k, meaning that the sequence is also identically distributed. The diagonal entries are strictly positive from the fact that a node always considers its current opinion. To check if (A3) is met just notice that in the complete case (3) is positive-symmetric. Hence the associated graph is undirected, complete and has therefore a globally reachable node. This result comes directly from the assumption of a complete underlying network. In the second case (4) the associated graph $\mathcal{G}^*$ will still be undirected and connected, thus presenting a globally reachable node.

Having convergence to consensus for (1), we are now interested in finding bounds on the rate of convergence. This would ensure that the stochastic system approaches the consensus value in a finite number of iterations in practice.

As investigated in [11], the mean-square convergence is particularly suited to analyze convergence in the setup of graphs with random topology. Given the sequence of stochastic averaging matrices $\{A(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ and corresponding solutions $x(k)$ to (1), the mean-square convergence factor is defined as:

$$\psi = \sup_{x(0) \neq x_{end}} \limsup_{k \to \infty} \left(\mathbb{E}\left[\left\| x(k) - \frac{1}{n} \mathbf{1}_n^T (x(k)) \mathbf{1}_n \right\|_2^2 \right] \right)^{\frac{1}{k}}$$

The following theorem provides upper and lower bounds for the mean-square convergence factor, which is analyzed in detail in [7].

Theorem 2 (Upper and lower bounds on the mean-square convergence factor)**.** *Under the same assumptions as in Theorem 1, the mean-square convergence factor satisfies:*

$$\psi \geq \rho_{ess}(\mathbb{E}[A(k)])^2$$

$$\psi \leq \rho\left(\mathbb{E}\left[A(k)^T \left(I_{nxn} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) A(k) \right] \right).$$

Here $\rho_{ess}(\mathbb{E}[A(k)])$ is the essential spectral radius of the row-stochastic matrix $\mathbb{E}[A(k)]$.

At this point, one may wonder if also the system (2) is asymptotically stable. This has been investigated in [6] obtaining the following result:

Theorem 3 (Consensus of expected value matrix)**.** *For a given random i.i.d. sequence $\{W(k)\}_{k \in \mathbb{Z}_{\geq 0}} = W(1), W(2), ...$ of stochastic matrices with positive diagonals, the following three statements are equivalent:*

(a) *The random sequence $\{W(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ is (weakly) ergodic almost surely.*

(b) *The deterministic discrete-linear dynamical system* $x(k) = \mathbb{E}[W(k)]x(k-1)$ *reaches consensus asymptotically.*

(c) $|\lambda_2(\mathbb{E}[W(k)])| < 1$.

Recall that in our case matrices in the sequence $\{A(k)\}_{k \in \mathbb{Z}_{\geq 0}}$ have strictly positive diagonals. Moreover, $\mathbb{E}[A(k)]$ is row-stochastic. This implies that it has an eigenvalue $\lambda_1 = 1$ and all the other eigenvalues strictly inside the unit circle, hence $|\lambda_2(\mathbb{E}[A(k)])| < 1$. In particular they will be real numbers in the case of (3) which is real-symmetric by construction. We can thus conclude system (2) reaches consensus asymptotically.

Model (1) describes a simplified version of the FPC algorithm that does not include an important feature, namely that every node can freely join the network in a permissionless setting. This is a critical feature that enables the network to be truly decentralized.

However, a permissionless network makes the job of the consensus algorithm a lot harder because bad actors could enter the network and, by collaborating, they could prevent it from reaching consensus on the valid transactions. For instance, modelling the bad actor as a stubborn node that does not update its opinion would imply convergence of the network opinion to the stubborn agent's state [8]. This can be easily understood noting that all other nodes will at some point sample the stubborn node and so their opinion will converge to the initial bad actor's opinion.

In reality, things can only get worse as not only one bad node could influence the opinion of the whole network but a sufficiently large group of malicious agents acting together could prevent the network itself from reaching consensus by voting against the majority opinion at each iteration. This could be modelled as a group of nodes that update their opinion according to a common time-varying strategy. In the worst case this would make the consensus mechanism really slow to terminate or even make the nodes not reach agreement on the termination value. The latter would imply that different honest nodes have different opinions on which transaction is valid and this in turn means that the ledger state is not consistent across the network anymore.

The FPC algorithm solves this problem by having nodes randomly choosing a threshold for each iteration that determines the minimum value for a node to change its opinion for the next iteration. This way the bad nodes cannot be sure of which opinion to advertise in order to make sure that the other nodes do not terminate.

This makes the problem a lot more complicated to approach from a control system point of view and so we introduced simplifications in order to make our system resilient to the influence of bad actors.

We will model a simplified scenario with a single stubborn node whose opinion is opposite to the initial majority opinion. Our goal is to make sure that our network is resilient to a stubborn agent by introducing a special node that we call *coordinator* which determines the value to which all nodes in the network will converge to. This might seem to make the network centralized, however, our model can be used also in case of many coordinators. This way, nodes that prove to be benign could be promoted to coordinators and incentives could be put in place to make sure that these nodes keep acting benignly. Therefore, bad nodes entering the network would have less power than the coordinators and so would not be able to influence its convergence.

In light of what we said above, we will consider three increasingly complex scenarios. First, let's consider the setup with n_c *coordinators* that apply a P-control strategy, and no stubborn agents. The P-controlled closed-loop dynamics can be written as follows:

$$x(k+1) = (\mathbb{E}[A(k)] - BK_P C)x(k) + BK_P r \qquad (8)$$

The following theorem is adapted by [4]. However, we propose a different and more generic proof valid for $n_c \geq 1$ which shows that the network achieves consensus at the n_c coordinators' reference signal r, when a low-gain controller is applied.

Theorem 4 (Hurwitz stability for P-controlled closed-loop systems)**.** *Consider the opinion-dynamics network model with* n_c *proportional controlled coordinators. Assume that the network graph G is strongly connected and aperiodic. For any sufficiently small gain* $(0 < k_P < \overline{k}_P)$*, the closed-loop state matrix is Hurwitz. Furthermore, the opinions of all agents converge asymptotically to the reference signal* r*.*

Proof. The state matrix for the opinion dynamics model with n_c P-controlled coordinators (8) is $A_P = \mathbb{E}[A(k)] - BK_P C$. Note that the assumption of G being strongly connected and aperiodic is equivalent to $\mathbb{E}[A(k)]$ being primitive. Recall also that $\mathbb{E}[A(k)]$ is row-stochastic. From Perron-Frobenius Theorem, we know that a primitive matrix has a real strictly positive, simple dominant eigenvalue λ and all the other eigenvalues μ satisfy $\lambda > |\mu|$. Hence, $\mathbb{E}[A(k)]$ has an eigenvalue in 1 and the rest strictly within the unit circle. By simply computing matrix products we can see that A_P is a perturbation of $\mathbb{E}[A(k)]$ by the following matrix term:

$$-k_P \begin{bmatrix} C \\ \mathbf{0}_{n_s \times n} \end{bmatrix}$$

For k_P sufficiently small, A_P keeps certain properties, namely A_P is still irreducible and thus non-negative, but it becomes row-substochastic. From this we can employ Corollary 4.13 from [5] which states that if a row-substochastic matrix is irreducible then it is convergent. Hence, A_P is Hurwitz and therefore (8) asymptotically converges to a given constant reference signal. This fixed point can be verified to be $r\mathbf{1}_n$ by checking that it is a solution for (8), i.e.:

$$r\mathbf{1}_n = (\mathbb{E}[A(k)] - BK_P C)r\mathbf{1}_n + K_P B r$$

From $\mathbb{E}[A(k)]\mathbf{1}_n = \mathbf{1}_n$ and $C\mathbf{1}_n = \mathbf{1}_n$, the result follows. ■

Remark 1: Note that A_P keeps the irreducibility property of $\mathbb{E}[A(k)]$ for sufficiently small k_P because each non-negative element of C has a corresponding non-negative element on $\mathbb{E}[A(k)]$. To see this recall that every edge of the graph associated to C is also an edge of the graph associated to $\mathbb{E}[A(k)]$ with a possibly different weight. Thus, for sufficiently small k_P every positive element of $\mathbb{E}[A(k)]$ will correspond to a positive element of $A_P = \mathbb{E}[A(k)] - k_P \begin{bmatrix} C \\ \mathbf{0}_{n_s \times n} \end{bmatrix}$.

Remark 2: It is important to understand that we do not really provide an upper bound of k_P for which the perturbed state matrix A_P is Hurwitz. Indeed we propose an interval $[0, \overline{k}_P])$ where we can leverage Corollary 4.13 and a quite restrictive sufficient condition for closed-loop stability when an arbitrary number of coordinators is considered. In our case we can state that a choice of:

$$\overline{k}_P = \frac{1}{p+1} \begin{cases} \frac{p}{n-1}, & \text{if network model is } \mathcal{G} \\ \min\limits_{i \in \mathcal{V}} \frac{p}{n_i - 1}, & \text{if network model is } \mathcal{G}^* \end{cases}$$

with n_i = $\mid \mathcal{N}_i \mid$ the out-degree of i, guarantees that A_P row-substochastic with n_c row-sums strictly less than 1.
However the largest possible $\overline{k}_P$ for which the eigenvalues of A_P are inside the unit circle can be computed using e.g. Routh-Hurwitz criterion on the perturbed state matrix or a discrete time root-locus analysis.

Now we consider the setup with the coordinators employing a PI-controlled strategy on their neighbours. The PI-controlled closed-loop dynamics can be represented as follows:

$$\begin{bmatrix} x(k+1) \\ w(k+1) \end{bmatrix} = \begin{bmatrix} \mathbb{E}[A(k)] - BK_PC & BK_I \\ -C & I_{n_c \text{x} n_c} \end{bmatrix} \begin{bmatrix} x(k) \\ z(k) \end{bmatrix} + \begin{bmatrix} BK_p \\ I_{n_c \text{x} n_c} \end{bmatrix} r\mathbf{1}_{n_c} \tag{9}$$

Unfortunately we were not able to prove the asymptotic stability of the above system (9) with an arbitrary number of coordinators. However we can leverage the following result from [4] when $n_c = 1$.

Theorem 5 (Hurwitz stability for PI-controlled closed-loop systems with only one coordinator)**.** *Consider the manipulated opinion-dynamics with a myopic proportional-integral controlled coordinator. If the network graph is strongly connected and aperiodic, then for any small combination of gains (*$0 \leq k_I \leq \overline{k}_I$, $0 \leq k_P \leq \overline{k}_P$*), the closed-loop system is Hurwitz stable. Moreover, the average opinion of the coordinator's neighbours will converge asymptotically to the reference.*

Having seen that one coordinator with PI strategy is indeed applicable to our system without making it unstable we would like to investigate if it is actually capable of eliminating, or at least reducing, the effect of malicious nodes. As shown in [10], consensus on the entire network will not be reached, but we are interested in weaker type of convergence, i.e. the value of the average opinion of the network.
The following theorem describes the scenario where a single coordinator is able to overcome the action of a stubborn agent. For a complete description and proof we address the reader to [4].

Theorem 6 (Convergence on coordinator's neighbours with stubborn agent)**.** *Consider a manipulated opinion-dynamics network model with PI-controlled myopic coordinator, which also has a stubborn agent. Assume that the network graph is strongly connected, and further there is a path from the coordinator to each measured neighbour upon removal of the stubborn agent. Then the coordinator is able to manipulate its measured opinion, in the sense that* $\lim_{k \to \infty} y(k) = r$.

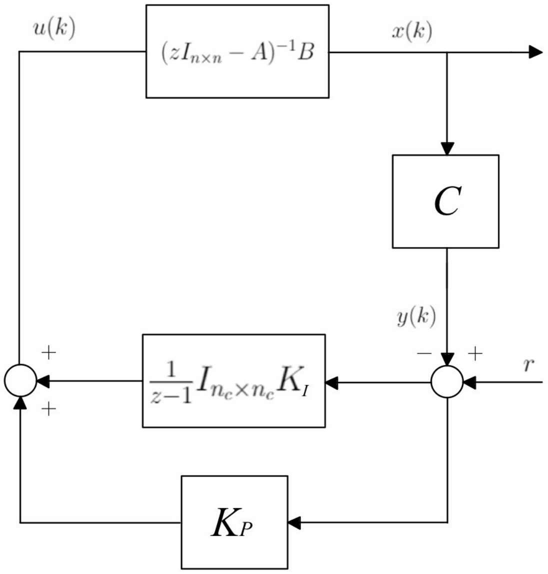
$u(k)$
$(zI_{n \times n} - A)^{-1}B$
$x(k)$
C
$y(k)$
$-$
$+$
r
$+$
$+$
$\frac{1}{z-1} I_{n_c \times n_c} K_I$
K_P

Fig. 1. Block diagram for the PI-controlled closed-loop system that models the manipulated opinion dynamics.

This implies that a coordinator is able to ”counterbalance” the action of the stubborn node on the honest part of the network, if we consider the mean consensus. However, the topology must satisfy some requirements. Intuitively, these assumptions are needed to make the coordinator able to influence directly the network and be more influential than the stubborn agent.

IV. Simulations and Numerical Results

Our previous analysis was based on the reasonable assumption that $\mathbb{E}[A(k)]$ is a good way to measure the evolution of a stochastic dynamical system. Therefore, we are interested in simulating the behavior of both (1) and (2) to verify that the state follows similar trajectories. We expect the asymptotic consensus value of (1) to change for a particular realization $\{A(k)\}_{k \geq 0}$, but to be centered, in expectation, about the consensus value attained by (2).
We will then provide an estimate of the worst case error of this approximation with respect to a reasonably large set of i.i.d. sequences $\{A(k)\}_{k \geq 0}$ drawn from $\mathcal{A} \times \mathcal{A} \times \cdots$.

Fig. 2 depicts the evolution of the opinion of a particular node for different values of p. Specifically, the opinion evolution derived by (2) and by five possible realizations of (1) are plotted. Since we have guarantees on almost sure consensus for the stochastic system under consideration coming from the network architecture and the assumptions made in the problem setup, it is not important which node's dynamics we choose to plot. Therefore, without loss of generality, we consider

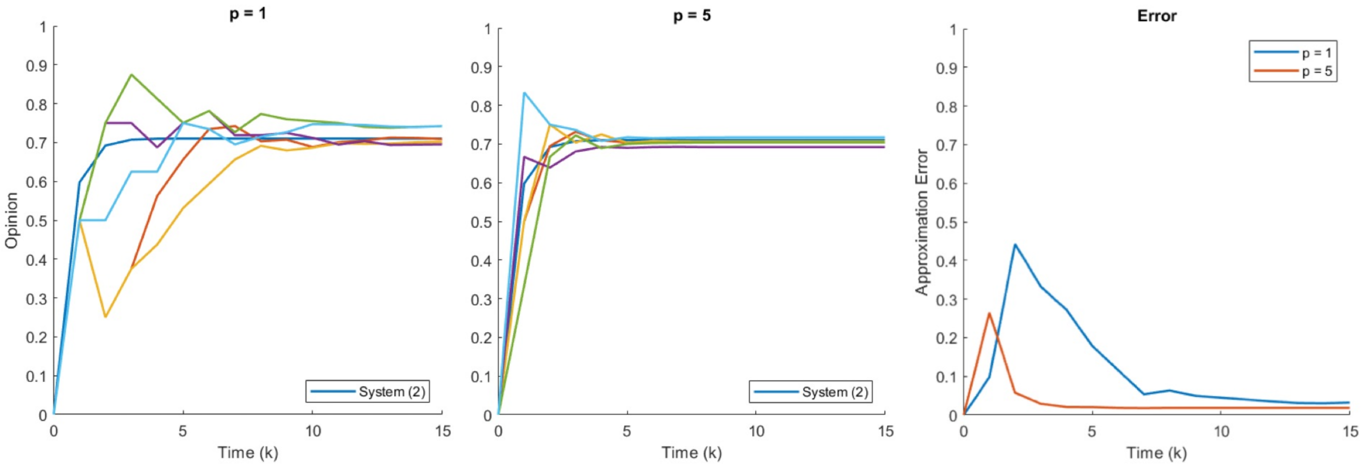

Fig. 2. Opinion evolution of an arbitrary agent for different p's. On the left, five random realizations of $\{A(k)\}_{k\in\mathbb{Z}_{\geq 0}}$ are plotted against the state trajectory of (2). The rightmost plot shows the magnitude of the largest error between the opinion dynamics of (2) and those of the same sequences of i.i.d. matrices at every time step.

the opinion evolution of a node starting with opinion 0 in a network where a clear majority of nodes has initial opinion 1. $\mathbb{E}[A(k)]$ becomes a finer approximation as more nodes are sampled. Indeed for larger values of p, the asymptotic consensus value of the stochastic system gets closer and closer to the one of (2), in expectation.

In addition, convergence to almost sure consensus seems to happen faster. Intuitively, someone's opinion converges to the average network opinion in a shorter times when he comes in contact with a larger number of external agents.

Notice that we will only consider $p < n-1$. The trivial case $p = n-1$ would mean that all agents are queried each round. This makes (1) become linear time-invariant. Since we have been dealing with properties of randomized averaging algorithms, this scenario would be too simple and of small interest for our analysis.

As explained in Theorem 1 the asymptotic behavior of a node i is given by the i-th entry of:

$$\lim_{k\to\infty} x(k) = (w^T x(0))\mathbf{1}_n$$

where $w \in \mathbb{R}^n$ is a random non-negative vector with $w_1 + w_2 + ... + w_n = 1$ such that:

$$\lim_{k\to\infty} A(k)A(k-1)\cdots A(0) = \mathbf{1}w^T$$

From the simulation, we can see that the almost sure consensus values for an arbitrary set of sequences $\{A(k)\}_{k\geq 0}$ are distributed about the one achieved by (2). Therefore, it seems that the final opinion of all nodes converges towards values close to:

$$\lim_{k\to\infty} \frac{\mathbb{E}[A(k)]^k}{\lambda^k} x(0) = vw^T x(0)$$

with $\mathbb{E}[A(k)]$ row-stochastic and primitive, $\lambda = 1$ the simple dominant eigenvalue, $v = \mathbf{1}_n$ and $w \geq 0$ respectively the right and left dominant eigenvectors of $\mathbb{E}[A(k)]$ such that $v^T w = 1$.

Remark 3: From Theorem 5.1 in [5] it can be noticed that the above equation yields the average of the initial network opinion in the case of a complete graph $\mathcal{G}$ for which (3) doubly-stochastic.

Thus, the stochastic model correctly describes the consensus behaviour of the FPC algorithm, in which the final network opinion will be the one corresponding to the dominant initial opinion. Just notice that all the random sequences reach consensus in a short time even when p is small, i.e. this randomized averaging algorithm is well-behaved also in the case of low message complexity. Thus (2) seems a reasonable approximation and a good measure of performance for the more general (1), providing also guarantees on the system's convergence rate (see Theorem 2).

However, one shortcoming of this model is that we are considering a clique in which each node is free to directly communicate with every other node. This is not always feasible because, in order to reduce network delay, nodes might be allowed to query only those that are geographically close. In the rest of this paper we will investigate the asymptotic behaviour of the feedback control systems where the coordinators are assumed to have limited visibility. Note that this comes with a price: the limited visibility model does not ensure that the global network average will also converge to the reference r because this is guaranteed only in the case of global visibility.

Specifically, the underlying network is chosen to be $\mathcal{G}^*$, with the following parameters:

(i) $n = 100$, number of nodes in the network;
(ii) $|\mathcal{N}_i| \in [p+1, \frac{n}{4}]$ for all $i \in \mathcal{V}$, chosen to have a sparse graph topology.

For what concerns the sampling process, we will consider $p = 5$. The initial opinion of each node, including the coordinators, is assigned by flipping a random coin biased 75% towards 1 in order to obtained a clear majority that we expect to be maintained by the controller. As in [9], we assign

to the stubborn node an opinion opposite to the initial majority. For the P-controlled system we fix $k_P = 10$, while for the PI formulation we set $k_P = 5$ and $k_I = 1$.
It is also worth mentioning that the more general scenario $\{A(k)\}_{k\in\mathbb{Z}_{\geq 0}} \sim \mathcal{A} \times \mathcal{A} \times \cdots$ will be considered instead of the manipulated opinion-dynamics in (6) and (7), which are already supported by mathematical results. In other words we want to show that the P and PI controllers can drive their neighbours' state close to the reference even when a randomized system is considered.

1) P-controlled system with $n_c = 2$ and $n_b = 1$: It has already been discussed in the proof of Theorem 4 that A_P is convergent for k_P small enough and arbitrary n_c. From now on we will assume the feedback system to be stable at every time step, i.e. we require stability in closed-loop for all $A(k) \in \{A(k)\}_{k\in\mathbb{Z}_{\geq 0}}$. In a certain sense, this is equivalent to have robustness to the uncertainty of the stochastic averaging system. Appropriate values for $\overline{k}_P$ can then easily be computed with a root-locus analysis in discrete time. It is also important to notice that the result in *Remark 2* only holds for (8) when no stubborn agents are modelled. Indeed asymptotic stability of all the closed-loops in the sequence $\{A_P(k)\}_{k\in\mathbb{Z}_{\geq 0}}$ is not guaranteed in this case.
However, if one malicious node is modelled some considerations on the state matrix in (8) based on its spectral radius can be done. From Theorem 4.11 and Corollary 4.13 in [5] we have the two following results for a matrix A_P that hold respectively in the non-negative and in the row-substochastic case:

(i) $\min\{A_P\mathbf{1}_n\} \leq \rho(A_P) \leq \max\{A_P\mathbf{1}_n\}$,
(ii) A_P is convergent if and only if there exists a directed path from each node i with $|\mathcal{N}_i^{out}| = 1$ to a node j with $|\mathcal{N}_j^{out}| < 1$.

For $0 < k_P < \overline{k}_P$, A_P is row-substochastic and hence $\min\{A_P\mathbf{1}_n\} < \max\{A_P\mathbf{1}_n\}$ and $\max\{A_P\mathbf{1}_n\} = 1$. Recall now that malicious nodes can only sample themselves, namely there is no directed path from them to the rest of the network. Therefore, the necessary and sufficient condition (ii) for convergence is not met, i.e. A_P has unitary spectral radius and an eigenvalue $\lambda = 1$ from the Perron-Frobenius Theorem.
Notice that A_P is block-triangular, as well as (5). Thus its eigenvalues will be the ones of the diagonal blocks $A_{P_{11}}$ and $A_{P_{22}} = I_{n_b\times n_b}$. In particular we have $\rho(A_{P_{11}}) < 1$ and eigenvalue $\lambda = 1$ semi-simple with algebraic multiplicity n_b. The first follows directly from irreducibility of the row-substochastic $A_{P_{11}}$, while the latter comes from the properties of the condensation digraphs with aperiodic sinks. Indeed, for row-stochastic matrices, as stated in Theorem 5.2 of [5], the eigenvalue 1 is semi-simple with multiplicity n_b if and only if all the sinks in the condensation digraph, regarded as subgraphs of $\mathcal{G}^*$, are aperiodic. In our case, all the diagonal elements of $A_{P_{22}}$ meet this specification from the problem setup. The state matrix of the P-controlled feedback (8) will therefore be semi-convergent and not convergent, meaning that perfect reference tracking cannot be achieved against stubborn agents. In particular we will witness a steady-state tracking error between the measurements and the reference.

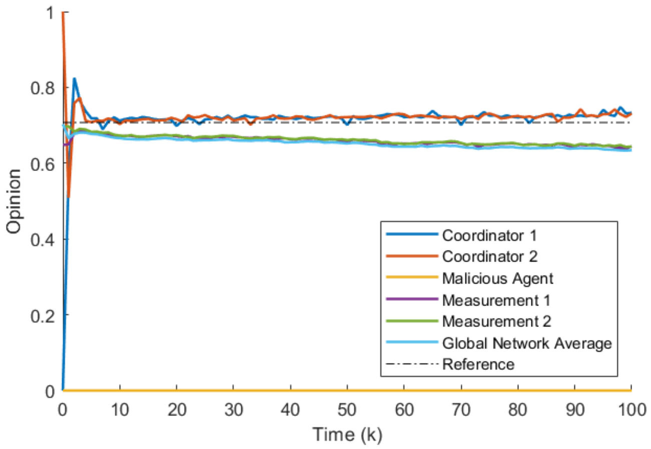

Fig. 3. The proportional controller cannot manipulate the network opinion for $k_P \in (0, \overline{k}_P)$, thus yielding a tracking error when k goes to infinity.

In our discussion we will consider $r = \textit{average}(x(0))\mathbf{1}_n$ because, in accordance with the FPC algorithm, we wish at least the average initial state to be maintained over time. In this particular plot, the coordinators start with opposite value but, after a few measurements (i.e. the $y_i(k)$'s defined in the problem setup) of the opinions of their neighbours , their own opinion quickly reaches a value higher than the reference so that they can balance out the influence of the stubborn agent. Indeed this is what we can see from Fig. 4, where the coordinators are capable of overcoming the stubborn node and steer the average opinion of their neighbours to the desired reference.

Remark 4: We now show experimentally that having convergence of the average opinion of the coordinators' neighbours to the reference r results in the convergence of the average global network opinion to a value which is close to r. As we can see from Fig. 5, the average network opinion reaches a value around r even if the coordinators cannot measure all the nodes of the network (myopic visibility) and with the presence of a stubborn node that tries to drive the network opinion to its own, which in this case is 0. This is a significant result because it shows that our system is able to counterbalance the stubborn node even if this was shown to be capable of strongly influencing the network opinion. Indeed, as explained in [8], without our control strategy, the malicious node would act as a leading node, steering the global network opinion to 0.
However, being able to keep the average network opinion close to the reference is an expected behaviour as, intuitively, the coordinators bring their neighbours' opinion to r, these in turn influence their own neighbours' opinion and so on, so that the network average remains to the desired value.

In the following simulations we will investigate if this intuitive reasoning still holds. In order to check this we will always plot both the coordinators' measurements and the global

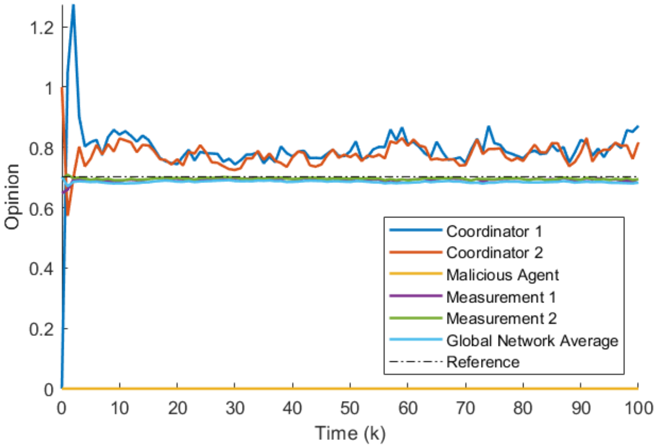

Fig. 4. Two P-coordinators with limited visibility (i.e. myopic manipulated dynamics) driving the arithmetic mean of their neighbours' opinion close to a constant reference r.

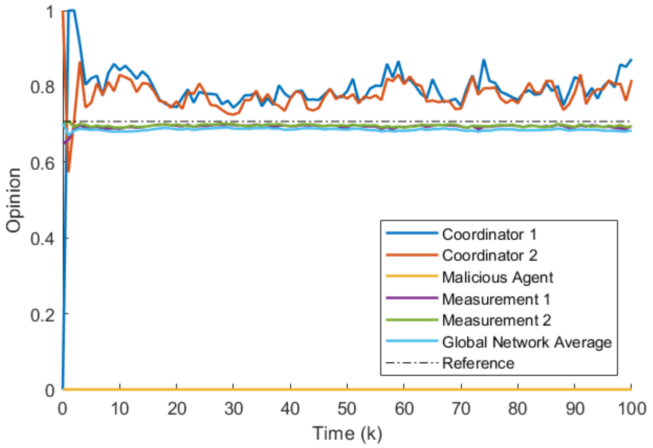

Fig. 6. Two coordinators with limited visibility driving the arithmetic mean of their neighbours' opinion to a constant reference r, with capping $[0, 1]^n$.

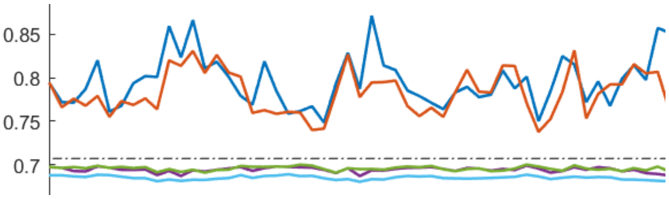

Fig. 5. Detail of Fig. 4 showing the tracking error of the measurements and the global network average with respect to. r.

network average and see if they appear to follow the same pattern. Note that this is a highly desirable behavior, because it ensures that a reasonable number of coordinators, with limited visibility and a common reference, can in principle steer the global behavior of a big network without reaching every node. Further, this implies that the coordinators themselves do not even need to know the topology of the whole network but just of a subset of it.

2) P-controlled system with saturation: We can see from Fig. 4, in which the P-control strategy is employed, that the opinions of the coordinators becomes greater than 1 for some time intervals. This happens because the coordinators try to overcome the action of the malicious node. In principle, the coordinators could set their state to arbitrarily large values in order to drive the average opinion closer to r. However this behaviour should not be allowed because the opinions on a transaction should belong to the closed interval $[0, 1]^n$. As shown in Fig. 6, even with this limitation, the system still manages to keep the output measurements close to r. Clearly, this saturating behavior is highly dependent on the choice of k_P: with a higher k_P, saturation is more likely.

Remark 5: It is important to understand that the guarantees of convergence from Theorem 4 for a sufficiently small k_P hold only when the system is deterministic and no malicious agents are present. For k_P's that are too small, i.e in range $(0, \overline{k}_P)$, where $\overline{k}_P$ is the one defined in *Remark 2*, the control action is not capable of maintaining the local and therefore the global network opinion close to the desired signal . Fig. 3 shows that the coordinator fails to do so for k_P approximately $\overline{k}_P$. On the other hand increasing k_p after a certain level makes the system less smooth and more likely to saturate. It seems that, in the presence of a stubborn node, there is an optimal interval of choice of k_P, i.e. $k_P \in [k_{P_{min}}, k_{P_{max}}]$, where $k_{P_{min}}$ and $k_{P_{max}}$ depend on the number of coordinators, the number of stubborn nodes and the network topology.

In conclusion the P-control strategy has some clear limitations and needs fine-tuning of the parameters for every specific case of application, thus in the rest of our discussion we will focus on the more effective control strategy employing a PI controller.

3) PI-controlled system with n_c = 2: The behaviour of the PI-controlled system is represented in Fig. 7. In this scenario we are not considering the malicious agent, as we are only interested in simulating the behaviour of the system (9) that was proved to be asymptotically stable by Theorem 6, but now we consider two coordinators instead of a single one. We conjecture that also in this case asymptotic stability is reached for sufficiently low k_P and k_I. Fig. 7 shows indeed that asymptotical stability is reached. Moreover we can notice how the behavior of the system with 2 PI-controllers is indeed smooth and this suggests us that the PI-strategy is probably better suited than the P-strategy for our case.

4) PI-controlled system with n_c = 1 and one stubborn agent: We now simulate the behavior of the PI-controlled system with saturation and one coordinator that tries to balance out a stubborn node. As can be seen in Fig. 8, the coordinator is able to counterbalance the action of the stubborn node on the other nodes in the network, if we consider the mean consensus. Theorem 6 proved that the average opinion of the coordinator's neighbours reaches the reference r but we didn't have any guarantees in the case with saturation and a time-varying system. Fig. 8 shows that both the measurement and the global network average remain bounded in a narrow interval about r for large k even if the coordinator's opinion saturates. This is

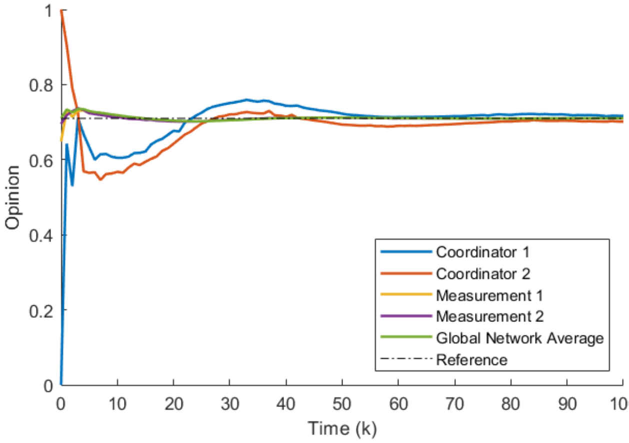

Fig. 7. The arithmetic mean of the two coordinators' neighbours' opinion is driven to a constant reference r.

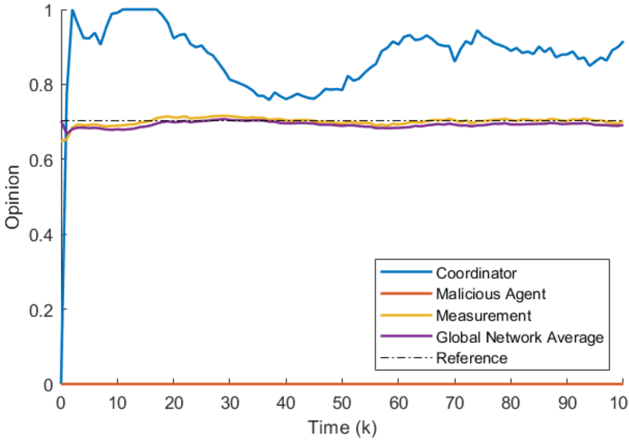

Fig. 8. One coordinator with saturation is still capable of maintaining the global network average to reference even in the presence of a stubborn node.

expected as the network is already starting with most nodes close to the reference (75% of nodes starting with opinion 1). Therefore, the coordinator only has to maintain the average opinion around the reference and for this it is not necessary to set its opinion to a value greater than 1.

What still needs to be discussed is an optimal placing of the coordinators in the network. So far they have been randomly chosen, but it would be interesting to explore more clever ways of selecting them, given their importance in guaranteeing the convergence of the network average to the desired behaviour. We propose to assign to a node the role of coordinator according to some centrality measures. These will give us a hint on the set of nodes that are more likely to influence the other nodes' opinion, so that it is easier for them to counterbalance the action of the stubborn node.
For instance, a measure of centrality that could be used to this end is degree centrality. This is defined as the number of edges incident upon a node, that is, the in-degree of a node. From [5], the degree centrality of node i is:

$$c_{degree}(i) = d_{in}(i) = \sum_{j=1}^{n} a_{ji} \tag{10}$$

where a_{ji} is the weight of the ji-th edge.

We tried this approach in the last scenario considered, i.e. a PI-controller and a stubborn node with saturation, and we were indeed able to improve our performance for what concerns the time of convergence. Specifically we considered how many time steps it took for the error between the global network average and the reference to reach a relative value of 2%. By choosing the coordinator to be the node of the graph defined by $\mathbb{E}[A(k)]$ with the highest c_{degree}, we were able to decrease the time of convergence by 17% compared to the case where the coordinator was randomly assigned.

V. Conclusion

In this paper, we proposed different models for the FPC consensus algorithm, each with its own simplifications that were necessary to make the system analyzable from a control system point of view. For each model, we either provided some mathematical guarantees on the convergence to consensus or we simulated its behaviour under various circumstances. We also considered the limitations of the proposed models and provided some insights on how to bypass these limits. As an example, we showed that consensus can be achieved even by averaging the opinions of nodes chosen with randomized sampling. However, we also showed that a single stubborn node is enough to influence the consensus value to its own opinion, which is not a good thing in a permissionless decentralized system. To overcome this limitation we introduced what we called coordinator nodes which are able to bring the average opinion of the network to a given reference. Unfortunately, this model is not resilient to other kinds of attacks on the network that may be done by more intelligent byzantine agents, which would require more sophisticated control strategy and mathematical tools. Finally, we have briefly investigated the use of centrality measures in assigning the role of coordinators to the nodes, to make them more influential on the rest of the network. By choosing the coordinators according to their centrality, we were able to obtain a faster convergence to consensus.

References

[1] S. Popov, “The Tangle”, IOTA Foundation, April 2018
[2] S. Popov, WJ Buchanan, “FPC-BI: Fast Probabilistic Consensus within Byzantine Infrastructures”, Journal of Parallel and Distributed Computing, 2021
[3] A. Capossele, S. Mueller and A. Penzkofer, “Robustness and efficiency of Leaderless Probabilistic Consensus Protocols within Byzantine Infrastructures”, IOTA Foundation, November 2019.
[4] N. Wendt, C. Dhal and S. Roy, “Control of Network Opinion Dynamics by a Selfish Agent with Limited Visibility”, IFAC-PapersOnline, Volume 52, 2019, Pages 37-42.
[5] F. Bullo, “Lectures on Network Systems”, Kindle Direct Publishing, 2019, ISBN 978-1986425643, with contributions by J. Cortés, F. Doerfler and S. Martìnez.
[6] A. Tahbaz-Salehi and A. Jadbabaie, “A Necessary and Sufficient Condition for Consensus Over Random Networks”, April 2008.

[7] F. Fagnani and S. Zampieri, “Randomized Consensus Algorithms Over Large Scale Networks”, IEEE Journal on Selected Areas in Communication, April 2008.

[8] J. Ghaderi, R. Srikant, “Opinion dynamics in social networks with stubborn agents: Equilibrium and convergence rate”

[9] A. Gogolev, N. Marchenko, L. Marcenaro and C. Bettstetter, “Distributed Binary Consensus in Networks with Disturbances”, ACM Trans. Auton. Adapt. Syst. 10, 3, Article 19, August 2015.

[10] E. Yildiz, A. Ozdaglar, D. Acemoglu, A. Saberi, A. Scaglione, “Binary opinion dynamics with stubborn agents”, ACM Transactions on Economics and Computation (TEAC) 1 (4), 1-30

[11] S. Silva Pereira, A. Pagès-Zamora, “Mean square convergence of consensus algorithms in random WSNs”, IEEE Transactions on Signal Processing 58 (5), 2866-2874