Distributed Detection of Malicious Attacks on Consensus Algorithms with Applications in Power Networks

Preprint

Sourav Patel,[1] Vivek Khatana,[1] Govind Saraswat,[2] and Murti V. Salapaka[1]

[1] *University of Minnesota*
[2] *National Renewable Energy Laboratory*

To be presented at the 2020 IEEE International Conference on Control, Decision and Information Technologies (CoDIT)
Virtual
June 29–July 2, 2020

NREL is a national laboratory of the U.S. Department of Energy
Office of Energy Efficiency & Renewable Energy
Operated by the Alliance for Sustainable Energy, LLC

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

Contract No. DE-AC36-08GO28308

Conference Paper
NREL/CP-5D00-76848
May 2020

Distributed Detection of Malicious Attacks on Consensus Algorithms with Applications in Power Networks

Preprint

Sourav Patel,[1] Vivek Khatana,[1] Govind Saraswat,[2] and Murti V. Salapaka[1]

[1] *University of Minnesota*
[2] *National Renewable Energy Laboratory*

Suggested Citation
Patel, Sourav, Vivek Khatana, Govind Saraswat, and Murti V. Salapaka. 2020. *Distributed Detection of Malicious Attacks on Consensus Algorithms with Applications in Power Networks: Preprint*. Golden, CO: National Renewable Energy Laboratory. NREL/CP-5D00-76848. https://www.nrel.gov/docs/fy20osti/76848.pdf.

Conference Paper
NREL/CP-5D00-76848
May 2020

National Renewable Energy Laboratory
15013 Denver West Parkway
Golden, CO 80401
303-275-3000 • www.nrel.gov

NOTICE

This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by Advanced Research Projects Agency-Energy under Grant DE-AR0001016. The views expressed herein do not necessarily represent the views of the DOE or the U.S. Government.

Distributed Detection of Malicious Attacks on Consensus Algorithms with Applications in Power Networks

Sourav Patel[1], Vivek Khatana[1], Govind Saraswat[2] and Murti V. Salapaka[1]

***Abstract*— Consensus-based distributed algorithms are well suited for coordination among agents in a cyber-physical system. These distributed schemes, however, suffer from their vulnerability to cyber attacks that are aimed at manipulating data and control flow. In this article, we present a novel distributed method for detecting the presence of such intrusions for a distributed multi-agent system following ratio consensus. We employ a Max-Min protocol to develop low cost, easy to implement detection strategies where each participating node detects the intrusion independently, eliminating the need for a trusted certifying agent in the network. The effectiveness of the detection method is demonstrated by numerical simulations on a 1000 node network to demonstrate the efficacy and simplicity of implementation.**

keywords: **Ratio consensus, distributed intruder detection, cybersecurity, smart microgrids, distributed algorithm.**

I. INTRODUCTION

Distributed coordination and decision making has enabled new paradigms in cyber-physical systems that include sensor networks, autonomous vehicle systems and power networks. A key strategy often employed is achieving consensus among the agents in a network by sharing local information with its neighbors [1], [2]. The applicability of Consensus-based distributed coordination methods often depends on the termination of the consensus algorithm in finite-time (see [3], [4] for static graphs and [5] for dynamic topology networks) which allows for the inferences reached to be used by the agents to perform subsequent important operations; for example, to provide ancillary services by local agents in a power grid (see [6] and [7]). Moreover, the implementation of such methods in a cyber-physical system might necessitate a prior centralized coordination step. In many situations, such centralized coordination is not feasible. To this end, a ratio consensus algorithm that eliminates the need of a centralized framework is presented in [8], [9]. Such distributed decision making schemes depend heavily on trust and credibility of the participating agents. Each agent gathers information by relying on it's neighbors to share their information honestly. When factoring this type of information exchange into a model, each agent's lack of honesty can skew the resulting decision (consensus value). It is imperative to increase the resilience of multi-agent systems towards such malicious or manipulative behaviour. To address the cyber-security issues in distributed algorithms, a number of different approaches have appeared in the literature. One such method encompasses trust based models for multi-agent systems [10]. Such trust-based approaches need repeated interactions between the agents and need to gather a large amount of information from different agents in the network to formulate notions of cooperation or rejection of transactions. Reference [11] formulates an intrusion detection system based on network traffic analysis whereas authors in [12], [13] assert the need of creating an r-robust graph for a resilient consensus. Thus, most proposed methods involve techniques that impose additional constraints on the network that are difficult to generalize.

This article presents a distributed algorithm for detection of malicious nodes in a cyber-physical system. We use the monotonic properties of global maximum and minimum over the network following the consensus algorithm, to detect intrusions. The proposed algorithm has many benefits over the existing counter parts in the literature [10]–[14]. Firstly, the scheme proposed eliminates the need of a unique identifier (as required in [14]) for each node in the network which allows each node to individually detect the intrusion. The results developed here enable a truly distributed way of detection as an agent utilizes information only from its neighbors. The only global information needed is an upper bound on the graph diameter D (defined in Section II). Furthermore, it obviates the need for any centralized planning beforehand. Secondly, the scheme proposed has a small memory requirement; specifically, each node needs to store D values for each of its neighbors (unlike the memory requirement which increases as a polynomial function of the number of nodes N in [14]). Also, the computations involved are simple mathematical operations like multiplication of scalar values, in contrast to techniques using matrix inversions and rank calculations which can easily become intractable as the size of the network increases. This makes the proposed algorithm suitable for *ad-hoc* networks where the agents have less computational power and storage capacity. Unlike other works (see [11], [14]) the proposed algorithm does not need a learning stage where the network parameters used for intrusion detection are estimated before the actual detection protocol can be deployed, thus, facilitating an on-line implementation in cyber-physical systems like smart microgrids. Also, the proposed scheme is topology independent unlike schemes where the graph topology is the determining

This work was authored in part by NREL, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the Advanced Research Projects Agency-Energy under Grant DE-AR0001016. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

[1]Sourav Patel {patel292}, Vivek Khatana {khata010} and Murti V. Salapaka {murtis}@umn.edu are with Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, USA. [2] Govind Saraswat {Govind.Saraswat@nrel.gov} is with National Renewable Energy Laboratory, Golden, CO, USA.

factor for tolerance towards malicious agents [13]. With the above benefits, this algorithm can be applied to a multitude of cyber-physical systems.

The rest of the paper is organized as follows. In Section II and III, we present the ratio consensus and distributed apportioning framework. In Section IV, different intruder models and their analysis are presented. Section V presents the proposed distributed detection scheme for creating a cyber-secured network. The proposed algorithm is validated on simulations and results are presented in Section VI. Section VII provides the conclusion and future directions.

II. Preliminaries

A. Definitions

In this section, we present basic notions of graph theory and linear algebra. Detailed descriptions are available in [15].

Definition 1. *(Directed Graph) A directed graph G is a pair (V, E) where V is a set of vertices (nodes) and E is a set of edges, which are ordered subsets of two distinct elements of V. If an edge from $j \in V$ to $i \in V$ exists then it is denoted as $(i, j) \in E$.*

Definition 2. *(Strongly Connected Graph) A directed graph is strongly connected if it has a directed path between each pair of distinct nodes i and j.*

Definition 3. *(Column Stochastic Matrix) A real $n \times n$ matrix $A = [a_{ij}]$ is called a column stochastic matrix if $0 \leq a_{ij} \leq 1$ and $\sum_{i=1}^{n} a_{ij} = 1$ for $1 \leq i, j \leq n$.*

Definition 4. *(Irreducible Matrix) A $N \times N$ matrix A is said to be irreducible if for any $i, j \in \{1, ..., N\}$, there exist $m \in \mathbb{N}$ such that $(\boldsymbol{A}^m)(i, j) > 0$.*

Definition 5. *(Primitive Matrix) A non negative matrix A is primitive if it is irreducible and has only one eigenvalue of maximum modulus.*

Definition 6. *(Diameter of a Graph) The diameter of a directed graph is the longest shortest directed path between any two nodes in the network.*

We will consider D as an upper bound on the diameter of the graph throughout.

Definition 7. *(Epoch) An epoch is defined as any D consecutive state update iterations i.e. k^{th} epoch defines the discrete-time interval $[kD + 1, (k + 1)D]$.*

Definition 8. *(In-Degree) In-degree of a node i is the number of elements $|N_i^-|$, where $N_i^- = \{j : (i, j) \in E\}$.*

Definition 9. *(Out-Degree) The out-degree of a node i is the number of elements $|N_i^+|$, where $N_i^+ = \{j : (j, i) \in E\}$.*

B. Ratio Consensus Protocol

Consider a directed graph $G = \{V, E\}$ containing N nodes. Each node $i \in V$ maintains two states at time k, denoted by $x_i(k)$ (referred as numerator state of node i) and $y_i(k)$ (referred as denominator state of node i). Node i updates its state at the $(k+1)^{th}$ discrete iteration according to the following update law:

$$x_i(k+1) = p_{ii}x_i(k) + \sum_{j \in N_i^-} p_{ij}x_j(k), \tag{1a}$$

$$y_i(k+1) = p_{ii}y_i(k) + \sum_{j \in N_i^-} p_{ij}y_j(k), \tag{1b}$$

where N_i^- is the set of in-neighbors of node i. We next present a result from [8] which establishes the convergence of the ratio $x_i(k)/y_i(k)$.

Theorem 1. *Suppose the weight matrix P with $P(i, j) = p_{ij}$ associated with the directed graph G is primitive and column stochastic with $P(i, i) > 0$ for all $i \in V$. Let $\{x_1(0), x_2(0), ..., x_N(0)\}$, $y_i(0) = 1$ be the initial conditions for numerator and denominator states respectively for all $i \in V$. Then, the ratio $\frac{x_i(k)}{y_i(k)}$ asymptotically converges to $\alpha := \frac{1}{N}\sum_{i=1}^{N} x_i(0)$ for all $i = 1, ..., N$ (referred to as ratio consensus).*

Lemma 1. *With the assumptions of Theorem 1, sum of state values of x and y are conserved, i.e. for any k,*

$$\textstyle\sum_{i=1}^{N} x_i(k) = \sum_{i=1}^{N} x_i(0), \;\; \sum_{i=1}^{N} y_i(k) = \sum_{i=1}^{N} y_i(0),$$

for updates according to (1a) and (1b).

Proof. Proof is straightforward and is omitted due to space constraints. □

C. Problem Formulation

Let q denote the intruder node violating the update rules (1a), (1b) such that, given $\rho > 0$, for all $i \in V$,

$$\left| \lim_{k \to \infty} \frac{x_i(k)}{y_i(k)} - \frac{\sum_{j=1}^{N} x_j(0)}{\sum_{j=1}^{N} y_j(0)} \right| > \rho, \tag{3}$$

$$\lim_{k \to \infty} \left(\frac{x_i(k)}{y_i(k)} - \frac{x_q(k)}{y_q(k)} \right) = 0. \tag{4}$$

In other words, the intruder's aim is to steer the final convergence value of the ratio consensus away from the true convergence value (ratio of sum of initial states) to a new desired value and achieve consensus among all the nodes in the network at the desired value.

Objective: *Given update rules (1a), (1b) for all $i \in V$ and an intruder q, the goal is to detect any intrusions of the form (3) and (4) in the network in a distributed manner.*

Next, we substantiate the motivations of intruders and the need for a distributed detection algorithm through a distributed power allocation framework. The proposed detection algorithm, however, is applicable for detecting malicious attacks in other consensus-based applications such as distributed optimization [16], [17], finite time consensus algorithms in higher dimensions [18], [19], movement coordination in autonomous vehicles, task allocation in unmanned aerial vehicles (UAVs) for rescue and search operations.

III. Framework for distributed apportioning in power networks

The ratio consensus protocol discussed above is used for distributed coordination of Distributed Generation (DG) units to meet the ancillary service demand. We first summarize below the distributed averaging protocol [7], [8], [20] Here the ratio consensus algorithm is used to compute the power reference commands for the DG units based on their generation capacities. Let ρ_d represent the total amount of power requested by an aggregator of the smart micro-grid to be supplied by the *N* DG units (see Fig. 1(a)). Let π_j^* represent the power supplied by DG *j* while respecting its own generation capacity and load constraints. π_j^{max}, π_j^{min} represent the maximum and minimum power the j^{th} DG unit can supply. Thus,

$$\sum_{j=1}^{N} \pi_j^* = \rho_d, \text{ and } \pi_j^{min} \leq \pi_j^* \leq \pi_j^{max} \text{ for all } j \in V. \quad (5)$$

We refer to the problem of determining π_j^* respecting the above constraints as the resource apportioning problem where (5) gives the feasibility criteria for all participating units. A fair way of apportioning DG units to supply the demanded power is given by [7]:

$$\pi_j^* = \pi_j^{min} + r(\pi_j^{max} - \pi_j^{min}) \text{ for all } j \in V, \quad (6)$$

$$\text{where,} \quad r = \frac{\rho_d - \sum_{j=1}^{N} \pi_j^{min}}{\sum_{j=1}^{N} (\pi_j^{max} - \pi_j^{min})}. \quad (7)$$

The power apportioning algorithm [7] provides a distributed way to calculate r when a single aggregator can communicate to at least one node in the network.

IV. Analysis of attack strategies

In this section, we describe various attack models for intruders. We classify the intrusion attacks based on two criteria: (1) the impact they have on the power network and (2) the kind of strategy used by these intruders. Based on the impact caused, intrusions are categorized as vandalism attacks and manipulative attacks. The vandalism attacks directed towards the power network are used to disrupt the proper functioning of the power network. Such attacks can be easily detected by monitoring physical quantities like voltage and current profile of the power network [21] and are not within the scope of this paper. Manipulation attacks on the other hand are difficult to detect. These attacks are responsible for diverting the power network from the actual operating point. Although, these attacks are not as severe as the vandalism attacks in their impact, the difficulty lies in detection of such attacks. A manipulation attack running for a long time may lead to large revenue losses.

Intruder model: We consider a class of intruders whose objective is to alter the consensus value by θ (called attack strength), viz. the consensus algorithm converges to $\alpha - \theta$ or $\alpha + \theta$ where $\alpha = \frac{\sum_{i=1}^{N} x_i(0)}{N}$. Without loss of generality, we will analyze the case of intruder trying to steer the consensus to $(\alpha - \theta)$ where $\theta > 0$.

A. Assumptions on intruder attack model

We provide the assumptions imposed on our attack model:

- **A1.** All the attacks are intended to manipulate the smart micro-grid performance (by manipulating the consensus value). The intruder is motivated to increase its profit by increasing its share of the total power supplied to the loads (similar to competitive Transactive Grids). Furthermore, we assume that attacks do not occur at $k = 0$.
- **A2.** The intruder has as much global information available to itself as every other agent in the network.

Based on the above assumptions we now provide various attack strategies:

1. Constant (False) data injection attacks: Here the motive of the attacker is to modify the consensus value by injecting a constant false value into the network. In such attacks the ratio of the state updates of the intruder node remains constant after some iteration k' i.e.

$$\frac{x_q(k)}{y_q(k)} = \frac{x^*}{y^*}, \quad (8)$$

for $k \geq k'$ where, q is the intruder node, and x^* and y^* are the intruder injected constant values. Note, that these kind of intrusion attacks also encompass stuck nodes. We extend the work of [22] requiring doubly-stochastic weight matrix and strong assumptions on the induced sub-graph after removing the intruder in the network be strongly connected. We present a theorem below which shows for the ratio consensus protocol all nodes in the network will converge to the value injected by the intruder.

Theorem 2. *With the attack strategy defined in (8), the ratio of numerator and denominator states $\frac{x_i(k)}{y_i(k)}$ for all nodes i converges to a common limit (consensus value). Further, the limit is equal to $\frac{x^*}{y^*}$.*

Proof. Proof is omitted due to page constraints. □

2. Data Manipulation attacks: Let T denote the set of discrete-time attack instants when an intruder injects malicious values in the network. Let the l^{th} attack by the intruder i occurs at instant t, then,

$$x_i(t+1) = \sum_{j \in N_i^- \cup i} p_{ij} x_j(t) - \delta_{k_l}, \quad (9)$$

where, $t \geq l$ and $\sum_{l \in T} \delta_{k_l} = N\theta$. State y_i is updated according to (1b). One such attack strategy is to choose δ_0 and a to achieve a desired attack, θ, within m number of attacks is,

$$\delta_0 = \left(\frac{1-a}{1-a^m}\right) N\theta, \;\; a \in [0,1], \quad (10)$$

with, $\delta_l = a^{l-1}\delta_0$ for some $\delta_0 > 0$.

The next theorem shows that the intruder has an attack strategy that can steer the ratio consensus protocol to achieve desired state in the network.

Theorem 3. *The intruder attack strategy given in (9) steers the consensus value to $(\alpha - \theta)$, where $\alpha = \frac{1}{N}\sum_{j \in V} x_j(0)$.*

Proof. (By Induction) Let α be the consensus value without attack. $\sum_{j\in V} x_j(0) = N\alpha$ (as $\sum_{j\in V} y_j(0) = N$). It is sufficient to prove that under the attack strategy (9) $\sum_{j\in V} x_j(k) = N\alpha - N\theta$. Consider i to be the intruder node and let $k_1, k_2, \ldots \in T$ be the attack instants where k_l denotes the l^{th} attack instant. From Lemma 1 with $l = 1$,

$$\begin{aligned}\sum_{j\in V} x_j(k_1) &= \sum_{\substack{j\in V\\ j\neq i}} \sum_{l\in N_j^-\cup j} p_{lj}x_j(k_1-1) \\ &\quad + \sum_{l\in N_i^-\cup i} p_{li}x_i(k_1-1) - \delta_{k_1} \\ &= \sum_{j\in V}\sum_{l\in N_j^-\cup j} p_{lj}x_j(k_1-1) - \delta_{k_1}\end{aligned}$$

(Induction Hypothesis) Let, for $l = m'$, $\sum_{j\in V} x_j(k_{m'}) = N\alpha - \sum_{l=1}^{m'} \delta_{k_l}$. For $l = m'+1$ and since no attack occurs between the instants $k_{m'}$ and $k_{m'+1}$, proceeding as above,

$$\sum_{j\in V} x_j(k_{m'+1}) = \sum_{j\in V}\sum_{l\in N_j^-\cup j} p_{lj}x_j(k_{m'+1}-1) - \delta_{k_{m'+1}}$$

(Using Lemma 1 and no attack between $k_{m'}$ and $k_{m'+1}$),

$$\begin{aligned}&= \sum_{j\in V}\sum_{l\in N_j^-\cup j} p_{lj}x_j(k_{m'}) - \delta_{k_{m'+1}} \\ &= \sum_{j\in V}\sum_{l\in N_j^-\cup j} p_{lj}x_j(k_{m'}-1) - \sum_{l=1}^{m'}\delta_{k_l} - \delta_{k_{m'+1}} \\ &= \sum_{j\in V}\sum_{l\in N_j^-\cup j} p_{lj}x_j(k_{m'}-1) - \sum_{l=1}^{m'+1}\delta_{k_l}.\end{aligned}$$

Therefore, induction holds. Hence, $\sum_{j\in V} x_j(k_m) = N\alpha - \sum_{\ell=1}^{m}\delta_{k_\ell} = N\alpha - N\theta$ .Thus, from [8] the ratio converges to $\lim_{k\to\infty}\frac{x_j(k)}{y_j(k)} = \frac{N\alpha - N\theta}{\sum_{j\in V} y_j(0)} = \alpha - \theta$. □

Remark 3.1. *Here, the attack strategy depends upon the knowledge of N (more generally $\sum_{j\in V} y_j(0)$) by the attacker node. During imperfect knowledge of the network, attacker can steer the consensus only to $\alpha - \frac{N'}{N}\theta$, where N' is an estimate of N by attacker.*

V. Cyber-secured distributed network framework

In this section, we propose a cyber-secured framework for networks running distributed consensus algorithms. We first present the max-min protocols to obtain global maximum and minimum values of the network and use the monotonicity property of these protocols [9] to devise the proposed distributed intruder detection algorithm. Define the maximum and minimum value of the ratio of numerator and denominator states given by (1a) and (1b) over all nodes at any time instant k as,

$$M(k) := \max_{i\in V}\frac{x_i(k)}{y_i(k)}, y_j(k)\neq 0, j\in V, \tag{11a}$$

$$m(k) := \min_{i\in V}\frac{x_i(k)}{y_i(k)}, y_j(k)\neq 0, j\in V. \tag{11b}$$

Lemma 2. *For all time instants $k' \geq k$ and for all $i \in V$,*

$$\frac{x_i(k')}{y_i(k')} \leq M(k),\ \frac{x_i(k')}{y_i(k')} \geq m(k). \tag{12}$$

Proof. See [9] for proof. □

The following theorem shows that the maximum (or minimum) value $M(k)$ (or $m(k)$) strictly decreases (or increases) after finite time.

Theorem 4. *Define the initial ratio vector as $r(nD) := \left[\frac{x_1(nD)}{y_1(nD)}\ \ \frac{x_2(nD)}{y_2(nD)}\cdots\frac{x_N(nD)}{y_N(nD)}\right]$ such that $\min(r(nD)) < \max(r(nD))$, where, $n = 0, 1, 2, \ldots$ Then,*

$$M((n+1)D) < M(nD) \text{ and } m((n+1)D) > m(nD)$$

Proof. See [9] for proof. □

A. Maximum-Minimum (MXP-MNP) Consensus Protocol

The Maximum and Minimum Consensus Protocol referred as MXP and MNP protocol computes the maximum and minimum of the given initial node values $v(0) = [v_1(0)\ v_2(0)\ \ldots\ v_N(0)]^T$ in a distributed manner. It takes $v(0)$ as an input and generates a sequence of node values based on the following updates for any node $i \in V$,

$$z_i(k+1) = \max_{j\in N_i^-\cup i} z_j(k),\ w_i(k+1) = \min_{j\in N_i^-\cup i} w_j(k), \tag{13}$$

where $z_i(0) = w_i(0) = v_i(0)$.

Lemma 3. *The estimates $z_i(k)$ and $w_i(k)$ in (13) converges to $\max_{j\in V} z_j(0)$ and $\min_{j\in V} w_j(0)$ respectively in finite time $k \leq D$ for all $i \in V$.*

Proof. See [3] for the proof. □

The MXP and MNP protocols are reset after every epoch and the initial conditions are set as the initial ratios held by the nodes, that is, $z_i(nD) = x_i(nD)/y_i(nD)$ and $w_i(nD) = x_i(nD)/y_i(nD)$ respectively for all $i \in V$. Notice that in the above scheme the global maximum and minimum values of the ratios at the beginning of an epoch are available to each node after next D iterations (as one iteration of the Max-Min algorithm is completed in D iterations) [3]. In other words $MXP(k) = M((k-1)D)$ and $MNP(k) = m((k-1)D)$.

B. Detection Algorithm

Based on the above discussion we now present a detection algorithm. The algorithm requires each node in the network to store and access the last D state updates of all of its neighbors. We define the values stored by a node i as: $x_{ij}[kD : (k+1)D]$ and $y_{ij}[kD : (k+1)D]$ for all $j \in N_i^-$. This storage is maintained in a $StorageBuffer$ at each node. If each node in the network has access to last D state updates of all of its neighbors, δ_m cannot be chosen successfully by intruder node i with the assumptions of **A1** and **A2** (following the attack strategy (9) and (10)) as it can be detected due to one of the following violations:

$$\frac{x_{ji}[kD : (k+1)D]}{y_{ji}[kD : (k+1)D]} \leq MXP(k+1) \text{ or,} \tag{14}$$

$$\frac{x_{ji}[kD : (k+1)D]}{y_{ji}[kD : (k+1)D]} \geq MNP(k+1), \tag{15}$$

for all nodes $j \in N_i^+$. Such a detection will happen simultaneously at all the neighboring nodes of i followed by the generation of an $AttackFlag$ signal that is emitted by each node to its neighbors. Within the next D iteration of the algorithm the $AttackFlag$ signal regarding intrusion is propagated to all the nodes in the network which results in the interruption of the consensus protocol and further actions can be taken to make the network secure. The complete detection algorithm is presented in Algorithm 1.

Remark 4.1. *The proposed detection algorithm requires each node i to store $|N_i^-| \times D$ values thus, can be implemented in low-cost devices such as Raspberry-pi units.*

Algorithm 1: Distributed intruder detection protocol (at each node $i \in V$)

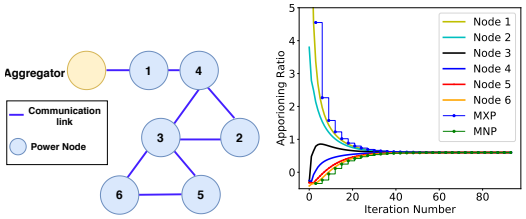
Repeat:
  Input:
    D;
  Initialize:
    $k = 1; l = 1$; $AttackFlag_i = False$;
    $StorageBuffer_i = [\,]$;
  Repeat:
    /* ratio consensus updates of states $x_i(k)$, $y_i(k)$ given by (1a) and (1b) */
    /* MXP and MNP protocol for each node $i \in V$ */
    $z_i :=$ MXP(k); $w_i :=$ MNP(k);
    (Using (11a) and (11b))
    $StorageBuffer_i \longleftarrow N_i^-$;
    /* Store last D values of all in-Neighbors. */
    if $k = lD$ **then**
      for $j \in N_i^-$ **do**
        if $\left\{\left(\frac{x_{ij}[1:(k-1)D]}{y_{ij}[1:(k-1)D]} - MXP(k)\right) \geq 0 \right.$ *or* $\left.\left(\frac{x_{ij}[1:(k-1)D]}{y_{ij}[1:(k-1)D]} - MNP(k)\right) \leq 0\right\}$ **then**
          $AttackFlag_i = True$;
          emit: $AttackFlag_i \longrightarrow N_i^+$;
          /* Send AttackFlag for termination. */
        else
          $StorageBuffer_i \longleftarrow [\,]$;
          /* Empty Buffer */
        end
      end
      $l = l + 1$;
    end
    $k = k + 1$;
  Output: $AttackFlag_i$ // Detection of intruder by node i

VI. Results

We now provide the validation of our detection scheme for a smart micro-grid involving distributed power apportioning as the cyber-physical system.

A. Simulation Results

In this section, we present simulation results to demonstrate distributed detection of an intruder attack in a power network. Intruder models developed in Section IV will be used to validate the detection strategy. Fig. 1(b) shows the state trajectories of the DGs and the $M(k)$ and $m(k)$ values following the distributed apportioning algorithm for communication network of Fig. 1 without any intruder in the network ($\alpha = 0.6$). The nodes have minimum and maximum power capacities as: $\pi^{min}(W) = [1000\ 1000\ 1000\ 1000\ 1000\ 1000]^T$, $\pi^{max}(W) = [1000\ 2000\ 5000\ 6000\ 4000\ 3000]^T$ with a total aggregator demand of 15 kW ($\leq \sum_{j \in V} \pi_j^{max}$). Fig. 2 presents simulation results for constant data injection attacks. Here, the malicious node 5 attacks the network by constantly injecting a ratio of 0.45 after iteration 30 ($\theta = 25\%$), emulating an actual aggregator demand of 12.75 kW as seen by the remaining (honest) nodes. Thus, the total power delivered by the rest of the power network becomes 10.4 kW, and node 5 can inject its maximum power (4 kW), making the total power delivered by the network as 14.4 kW. It is important to note that aggregator cannot distinguish between the two scenarios where the power network's total capacity is just 14.4 kW (deficit) or the case where attack is causing the total power delivered to be 14.4 kW. It can be observed that the intruder node is able to steer the consensus to its desired ratio (0.45 in this case) in the absence of any detection strategy. However, in the presence of the detection algorithm, intrusion is detected when node 5 violates Lemma 2 in the next $2D$ iterations (iteration number 36) resulting in detection ($AttackFlag = 1$) (Fig. 2). The following results are obtained by running simulations for a set of 10,000 graphs with 1000 nodes with a maximum possible diameter of 40. At each run we pick a graph with arbitrary initial conditions for all nodes. The attack strength (θ) is varied from 1% to 90% of the consensus value (α). We created



(a) (b)

Fig. 1: (a) Communication topology of an associated power network interfaced with an aggregator ($D = 3$) and node 1 as the demand circulation node, (b) state value updates with MXP and MNP values over the network.

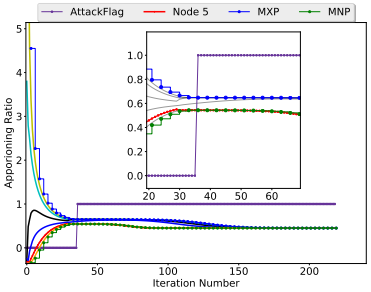


Fig. 2: A 6-node power network with the malicious node 5 launching a constant injection attack resulting in detection.

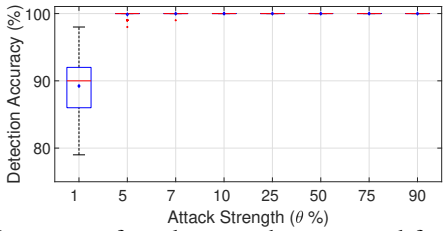


Fig. 3: Instances of random graphs generated for 1000 node network for attack statistics.

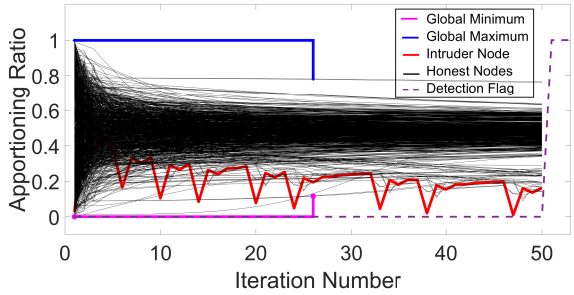


Fig. 4: Malicious node launching data manipulation attack in a 1000-node power network. Detection at iteration 50 $(2D)$.

30 million possible scenarios by varying the graph topology, attack strength ($\theta\%$ of the consensus value α), intruder node, time instant of attack, number of attacks. The detection accuracy of the proposed algorithm for all cases are shown in Fig. 3.

Remark 4.2. *Our algorithm detects smaller attacks ($\theta < 5\%$) with an average accuracy of 89% which are harder to detect. In such cases the intruder cannot steer the final consensus to achieve larger deviations. Medium attacks ($5 \leq \theta \leq 10\%$) are detected with an accuracy $> 99.5\%$ where as stronger attacks ($\theta > 10\%$) are always detected.*

Remark 4.3. *The outliers (denoted by red crosses in Fig. 3) in the simulations that avoided detection for smaller and medium attack strengths, were mostly observed to be cases where the intruder could inject a large value when the range of minimum and maximum initial values across all the nodes in the network were large. However, such a strategy doesn't guarantee success in attack as intruder has no knowledge of this range in this framework.*

An example case of data manipulation attack is shown in Fig. 4. The global maximum and global minimum are shifted by D iterations to the left for clarity but are obtained only after D iterations in the protocol. Here the intruder node with an attack strength of 3% following (9) and (10) with $a = 0.95$, $m = 95$ and $\delta_0 = 7.3 \times 10^{-4}$. However, it violates the MNP criteria in (14) leading to detection and sending $AttackFlag = 1$ at iteration number 50 $(2D)$.

VII. Conclusion and Future Work

In this work, we considered the problem of detecting malicious attacks in distributed cyber-physical systems. Attack models for intruders were proposed and analyzed that have not been previously considered in the literature. A distributed intruder detection algorithm was presented, analyzed and validated using simulations. The proposed detection method makes use of only local information of the nodes under limited storage and computation requirements as opposed to existing detection methods. Analyzing the efficacy of the proposed scheme under time varying topologies and web-socket based experiments on a communication network are future directions for the current work.

References

[1] J. N. Tsitsiklis, "Problems in decentralized decision making and computation." DTIC Document, Tech. Rep., 1984.

[2] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *Proc. of IEEE CDC*, New Orleans, 2007, pp. 5492–5498.

[3] V. Yadav and M. V. Salapaka, "Distributed protocol for determining when averaging consensus is reached," in *45th Annual Allerton Conf*, 2007, pp. 715–720.

[4] N. E. Manitara and C. N. Hadjicostis, "Distributed stopping for average consensus using double linear iterative strategies," in *Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on*. IEEE, 2014, pp. 739–746.

[5] G. Saraswat, V. Khatana, S. Patel, and M. V. Salapaka, "Distributed finite-time termination for consensus algorithm in switching topologies," *arXiv preprint arXiv:1909.00059*, 2019.

[6] A. D. Dominguez-Garcia and C. N. Hadjicostis, "Distributed algorithms for control of demand response and distributed energy resources," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. IEEE, 2011, pp. 27–32.

[7] S. Patel, S. Attree, S. Talukdar, M. Prakash, and M. V. Salapaka, "Distributed apportioning in a power network for providing demand response services," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct. 2017, pp. 38–44.

[8] A. D. Dominguez-Garcia and C. N. Hadjicostis, "Coordination and control of distributed energy resources for provision of ancillary services," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. IEEE, 2010, pp. 537–542.

[9] M. Prakash, S. Talukdar, S. Attree, S. Patel, and M. V. Salapaka, "Distributed Stopping Criterion for Ratio Consensus," in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Oct. 2018, pp. 131–135.

[10] S. P. Marsh, "Formalising Trust as a Computational Concept," Ph.D. dissertation, University of Stirling, 1994.

[11] M. Toulouse, H. Le, C. V. Phung, and D. Hock, "Defense Strategies against Byzantine Attacks in a Consensus-Based Network Intrusion Detection System," p. 16, 2017.

[12] H. Zhang and S. Sundaram, "Robustness of information diffusion algorithms to locally bounded adversaries," in *American Control Conference*, Jun. 2012, pp. 5855–5561.

[13] J. Usevitch and D. Panagou, "Resilient leader-follower consensus to arbitrary reference values," in *American Control Conference*, 2018.

[14] S. Sundaram and C. N. Hadjicostis, "Distributed Function Calculation via Linear Iterative Strategies in the Presence of Malicious Agents," *IEEE Transactions on Automatic Control*, vol. 56, no. 7, Jul. 2011.

[15] R. Diestel, *Graph theory (graduate texts in mathematics; 173)*. Springer-Verlag Berlin and Heidelberg GmbH & amp, 2000.

[16] V. Khatana, G. Saraswat, S. Patel, and M. V. Salapaka, "Gradient-consensus method for distributed optimization in directed multi-agent networks," *arXiv preprint arXiv:1909.10070*, 2019.

[17] V. Khatana and M. V. Salapaka, "D-distadmm: A $o(1/k)$ distributed admm for distributed optimization in directed graph topologies," *arXiv preprint arXiv:2003.13742*, 2020.

[18] J. Melbourne, G. Saraswat, V. Khatana, S. Patel, and M. V. Salapaka, "On the geometry of consensus algorithms with application to distributed termination in higher dimension," *in the proceedings of International Federation of Automatic Control (IFAC)*, 2020.

[19] ——, "Monotonicity of consensus in high dimension and a peer to peer convex hull algorithm," *Preprint*, 2020.

[20] M. Prakash, S. Talukdar, S. Attree, V. Yadav, and M. V. Salapaka, "Distributed stopping criterion for consensus in the presence of delays," *IEEE Transactions on Control of Network Systems*, 2020.

[21] M. Liao and A. Chakrabortty, "Optimization algorithms for catching data manipulators in power system estimation loops," *IEEE Transactions on Control Systems Technology*, no. 99, pp. 1–16, 2018.

[22] R. Gentz, S. X. Wu, H. Wai, A. Scaglione, and A. Leshem, "Data injection attacks in randomized gossiping," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 523–538, Dec 2016.