A Cyber-Physical Systems-Based Checkpoint Model for Structural Controllability

Cristina Alcaraz, Javier Lopez

Computer Science Department, University of Málaga, Spain

{alcaraz,jlm}@lcc.uma.es

December 23, 2018

Abstract

The protection of critical user-centric applications, such as Smart Grids and their monitoring systems, has become one of the most cutting-edge research areas in recent years. The dynamic complexity of their cyber-physical systems (CPSs) and their strong inter-dependencies with power systems, are bringing about a significant increase in security problems that may be exploited by attackers. These security holes may, for example, trigger the disintegration of the *structural controllability* properties due to the problem of non-locality, affecting, sooner or later, the provision of the essential services to end-users. One way to address these situations could be through automatic checkpoints in charge of inspecting the healthy status of the control network and its critical nature. This inspection can be subject to special mechanisms composed of trustworthy cyber-physical elements capable of detecting structural changes in the control and activating restoration procedures with support for warning. This is precisely the aim of this paper, which presents a *CPSs-based checkpoint model* with the capacity to manage heterogeneous replications that help ensure data redundancy, thereby guaranteeing the validity of the checkpoints. As a support to this study, a theoretical and practical analysis is addressed to show the functionality of the approach in real contexts.

Keywords: Cyber-physical systems, critical control systems, structural controllability, Smart Grids.

1 Introduction

One example of today's typical user-centric applications is the Smart Grid, in which end-users benefit from the efficient provision of primary electrical supplies. The production and distribution of these goods to end-users are monitored 24/7 by specialised monitoring and supervision systems. These systems connect remote substations together to control all those cyber-physical systems (CPSs) that are integrated as part of the Smart Grid and its physical entities, such as generators, transformers or transmission pylons. In this context, a CPS embraces a set of autonomous and intelligent devices (e.g. sensors, actuators, controllers, gateways, servers, smart meters or robots) capable of (locally or remotely) monitoring and managing data flows and operations (e.g. measurements or commands), and supervising the operational performance at all times. Their actions are totally collaborative and can be supported by diverse types of technologies, from wireless technologies (e.g. wireless industrial sensor networks or MANETs) with support for Internet connection through IPv6 or 6LowPAN (RFC 6272, Internet Protocols for the Smart Grid [1]), to cloud computing technology to guarantee data centralisation and backup.

However, the use of CPSs for the protection of large critical systems might also bring about numerous security problems, probably caused by the integration of multiple computation, communication and physical elements as stated by Pasqualetti *et al.* in [2]. Through their conceptual models it is possible to understand the gravity of the exposure of a critical system to diverse types of faults or attacks. These may even hamper the provision of resources and essential services to end-users and may have irreversible effects due to the existing (inter)-dependencies of the underlying subsystems. This is the reason that both public and private entities are becoming involved in many aspects of defence and its influence on critical sectors, in which control is a key target in the sights of potential attackers. Concretely, government entities have been reporting the number of incidents over the last few years, such as the European Union Network and Information Security Agency (ENISA) in [3] and the Industrial Control System Cyber Emergency Response Team (ICS-CERT) in [4]. Their annual reports clearly show the rate of incidents in the different critical sectors, exposing the sensitivity of the energy sector to certain exploitation derived from information and communications technologies, such as a denial of service and corruption of physical and cyber resources [5].

To comply with an acceptable protection level, this paper presents a fault and intrusion detection approach to protect the structural properties of those user-centric applications whose monitoring systems and topologies tend to be quite susceptible to diverse types of variations, either caused by an adversarial influence or unintentional disturbances. The approach is based on: (i) a collaborative network composed of a subset of trustworthy elements responsible for inspecting the control and its critical nature, and on a data replication model to guarantee backup copies [6, 7, 8, 9].

According to Ruchika in [6], heterogeneous replication-based monitoring approaches guarantee tamper-resistance in critical infrastructures in general, by introducing diversity into monitoring replicas; a theory also sustained by Veronese *et al.* in [8]. Namely, through these approaches it is possible to maintain replicates of critical data in several parts of the system. In this way if one part of the system goes down or becomes inaccessible, there are other, possible paths to reach the data itself and discover why. However, large control distributions also need to be modelled, applying the technical capacities related to *structural controllability* given by Lin in [10], and whose concept is supported by the POWER DOMINATING SET (PDS) problem introduced by Haynes *et al.* in [11]. These last authors were motivated in part by the structures of electric power networks and the need to obtain an efficient monitoring of their systems. The representation of both concepts is done in this paper through graph theory, thereby illustrating the behaviour of large control networks, and evaluating the effectiveness of the approach in the face of diverse types of threats.

The paper is structured as follows: Section 2 introduces the preliminary concepts principally related to structural controllability and PDS, as well as the contextual conditions associated with the application context to be analysed. Section 3 defines the system model, the adversarial model and the threat scenarios so as to later describe the detection approach in Section 4 together with a theoretical and practical study to validate it. Finally, the paper concludes in Section 5 and outlines future work.

2 Preliminary and Contextual Conditions

As the implementation of large cyber-physical control networks can become costly, an easy and cheap way to do it could be through structural controllability. Structural controllability is an evolution of the traditional control theory expounded by Kalman in [12] in the 60s, the concept of which can be modelled according to the following time-dependent linear dynamical system:

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t), \qquad x(t_0) \ = \ x_0 \tag{1}$$

From this formulation [13, 14], $x(t)$ comprises a vector $(x_1(t),\ldots,x_n(t))^T$ holding the current state of a system with n nodes at time t; A illustrates a matrix $n \times n$ represeting the network topology; B corresponds to an input matrix $n \times m$ where $m \leq n$ identifies the set of nodes controlled by a input control vector $u(t) = (u_1(t),\ldots,u_m(t))$ containing independent signals [15]. To ensure the controllability of this equation, Kalman's rank criterion has to be considered such that rank$[\mathbf{B},\mathbf{AB},\mathbf{A}^2\mathbf{B},\ldots,\mathbf{A}^{n-1}\mathbf{B}] = n$. However, the computation of this rank may become prohibitive for those applications whose networks grow day by day through the incorporation of new devices, as is the case of Smart Grid systems.

An alternative to this problem is structural controllability [10], which analyses and depicts the properties of control and their relationships following graphical formulations given by graph theory [16]. Specifically, it focuses on providing the concept of 'controllability' through graph-based structures of type $\mathscr{G}(\mathbf{A},\mathbf{B}) = (V,E)$. In this graphical-theoretical interpretation, $\mathscr{G}$ is a digraph without cycles that illustrates the direction of control with $V = V_{\mathbf{A}} \cup V_{\mathbf{B}}$ representing the set of vertices and $E = E_{\mathbf{A}} \cup E_{\mathbf{B}}$ the set of edges; equivalent to say, for example, a set of control devices (V) such as sensors, servers or actuators, and a set of communication links (E). Likewise, $V_{\mathbf{B}}$ comprises all those nodes capable of injecting control signals throughout the network [15]; so $V_{\mathbf{B}}$ contains the nodes $\{(u_1(t),\ldots,u_m(t))\}$ from Equation 1.

To extract the minimal set of nodes $V_{\mathbf{B}}$ in $\mathscr{G}(\mathbf{A},\mathbf{B}) = (V,E)$ from a given $\mathscr{G}(V,E)$ and illustrate a graphical scheme based on driver nodes ($V_{\mathbf{B}}$) and observed nodes ($V_{\mathbf{A}}$), it is first necessary to apply, either the PDS problem or the maximum matching problem for bipartite digraphs [14]. Both techniques go through the entire graph analysing node-by-node, the degree of 'dominance' that these nodes have with respect to their neighbourhood. The result of the operation leads to two important sets, the minimum subset of driver nodes (denoted here by N_D) and the observed nodes, which are controlled by at least a driver node; i.e. $O \longleftarrow V \setminus N_D$. However, as the PDS was created in relation to the structures of the energy systems and their monitoring systems and, in addition, some authors [13, 17] have already proven the validity of the matching technique to extract N_D, our research focuses on the PDS problem.

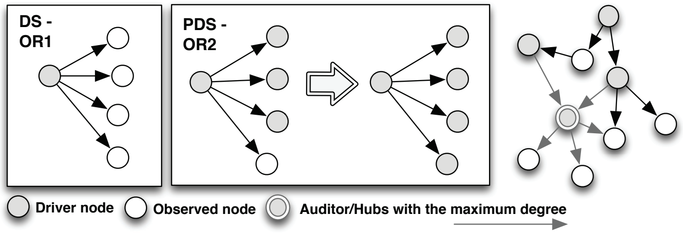

Figure 1: Nodes: Controllers, observed nodes and auditors

Conceptually, the extraction of N_D through the PDS is principally based on two fundamental observation rules, which were originally defined by Haynes *et al*. in [11] but later simplified by Kneis *et al*. in [18]:

OR1 *A vertex in N_D observes itself and all its neighbours*, complying with DS. The result is a new set of nodes named **OR1** as depicted in Figure 1.

OR2 *If an observed vertex $v_i \in$ **OR1** of degree $d^+ \geq 2$ is adjacent to $d-1$ observed vertices, the remaining unobserved vertex becomes observed as well* (see Figure 1).

From this definition it is possible to determine that **OR1** is contained within the definition of **OR2**, such that the subset of nodes that comply with **OR1** is part of the subset of nodes that comply with **OR2**. This also means that the control depends on the compliance of both rules, and any topological change can also signify a fault in the fulfilment of **OR1-2**, and therefore the degradation of the system. Also note that although these two rules are denoted as observation rules, we apply them here to the dual problem related to controllability.

Lastly, the applicability of structural controllability in general-purpose networks (e.g. random networks) is not enough to model critical applications. It is also necessary to determine the type of network to be built as part of the control structures and its underlying infrastructures. Monitoring networks deployed in critical scenarios tend to concentrate their networks in small substations in which the control is centralised in some devices (e.g. remote terminal units or gateways) and whose information is later centralised in the main control servers. The result is a set of subnetworks producing decentralized topologies with similar degree sequences to $y \propto \beta x^{-\alpha}$ [19] such as power-law out-degree (PLOD) [20]) and Barabási-Albert (BA) [21] (also known as scale-free distributions).

3 The Adversarial Model and Threat Scenarios

Taking into account the contextual conditions and the preliminary concepts introduced in the previous section, three types of node sets comprise our approach as also shown in Figure 1

- The driver nodes $\in N_D$ such that $\mid N_D \mid \geq 1$.
- The observed nodes included within the set O but $\notin N_D$, and controlled by at least one member $\in N_D$ such that $\mid O \mid \geq 0$. Note that according to the definition of **OR1** a node $\in N_D$ does not belong to O given that a driver node can be observed by itself. Likewise, those nodes $v_i \notin O$ are considered unobserved nodes belonging to U i.e. the set of unobserved nodes.
- The group of nodes serving as network auditors, denoted here by A^+. These auditors are responsible for inspecting the network conditions and its structural controllability such that $\mid A^+ \mid \geq 1$. The assignation of auditors will depend on two criterion given below whose dynamic re-assignation will depend on the degree of degradation of the network and the number of available driver nodes within the network.

These three kinds of nodes have the ability to autonomously process information and collaboratively interact with each other. Each node has assigned to it a unique identifier (ID) whose value is constrained to recognised identifiers; i.e. $\mathrm{ID}_i \in [X \ldots Y]$. We assume that the communication between peers includes the minimal conditions for establishing a suitable communication, such as synchronisation and the use of a time-stamp in each message with a unique counter.

The adversary model follows a weak model in which adversaries may be able to access the general structure of the graph, its topology and its driver nodes in order to exploit existing vulnerabilities. Here it is assumed that the attacker's mobility is limited to a random subset of $\delta \leq \frac{|V|}{2}$ nodes, where the attack models primarily focus on exploiting population homogeneity as defined in [6]. This strategy attempts to perturb the communication by randomly manipulating the maximum number of (controllers and observed) nodes, thus violating the availability of the medium access, the integrity of the communications and the topology of the network. Apart from this assumption, we also take into account that a change in the structural controllability can also come from an accidental perturbation (unforeseen alterations or malfunctions) or a high dynamism within the network (e.g. the use of smart phones, robots or tablets). We therefore specifically look at dynamic scenarios and threats related to crashes or physical attacks [6] that may (unintentionally or deliberately) corrupt the two observation rules defined above: **OR1** and **OR2**.

Depending on the type of attack/fault, we find several types of threat scenarios (denoted as SCN): [**SCN-1**] randomly removes a few (not all) edges of one or several vertices, which may (i) compromise the controllability of those dependent nodes or (ii) disconnect parts of the control; [**SCN-2**] randomly isolates one or several vertices from the network by intentionally deleting all their links; i.e. this attack may result in the complete isolation of nodes from the network; [**SCN-3**] randomly adds a few (but not all) edges to one or several vertices, which may increase the degree in the power-law distribution and alter the structural controllability. In real applications, it is important to consider authentication mechanisms since any new edge can signify the existence of a new member in the network or the illicit incorporation of a new link; [**SCN-4**] a combination of threats, which includes **SCN-1**, **SCN-2** and **SCN-3**. These scenarios, probably resulting from an attack, a simple crash or from the joining/leaving of a legitimate node, may take place at any moment of the execution of a system process, which might temporarily or permanently leave the operational tasks in an inoperative state.

One way to prevent anomalous situations would be through monitoring and inspection systems with the ability to replicate evidence for fault and intrusion detection. For the monitoring activities, a set of auditors has to be addressed, in which the assignation of their roles does not necessarily remain the same over time. This is due to the assumptions given in [22] about repair of controllability, which are simplified with the following protection condition (PC) required for our approach:

[**PC-1**] *Upon any suspicion of a possible change in the network structure or violation of the structural controllability, the system must always restore the network parameters*. Note that this process may entail the partial or total re-computation of the two elementary observation rules, **OR1** and **OR2**, specified in [23].

Lastly, the replication-based diagnosis process also forces us to consider (not now, but later) a new condition for the system model: $\forall v_j, v_i \in V$ such that $(v_i, v_j) \in E$, $\exists$ a m_{v_i} representing a critical alarm with information about those IDs of the graph that have misbehaved within the structural controllability. Depending on the criticality of the threat, the propagation of this alarm can be done in multicast mode to ensure data redundancy between the neighbours located at one-hop, or in unicast mode to connect with the main auditors of the system. Note that at this point we differentiate between internal auditors and main auditors. The latter correspond to the roots of control which are responsible for visualising the general state of the entire system and warning the main entities, such as the central system or human operators in the field, of the situation.

4 Checkpoint Model: Analysis and Validation

Taking as a reference the work done in [24, 25] on checkpoints and the protection condition (**PC-1**) defined in Section 3, our checkpoint model is based on a heterogeneous replication-based monitoring approach and on a finite set of trustworthy auditors $A^+ = [a_1, a_2, \ldots, a_n]$, such that each $a_i \in A^+$ also belongs to N_D. This trust level, essential to limit the scope of the approach, means that the nodes are a priori uncompromising devices since they are part of the organization and the control system. Given this, the construction of A^+ is determined by:

B^{A+}_{cond1} Monitoring entails a hierarchical structure in which the control is generally centralised in the roots of the graph $\mathscr{G}$. This means that the control direction and the audit edges in $\mathscr{G}$ always end in the roots of the network.

B^{A+}_{cond2} As the underlying structure follows a degree distribution, we select the set of auditors $\in N_D$ according to those driver nodes with the maximum degree (i.e. the hubs, by which the main control loads pass through them such as RTUs, gateways or servers).

At this point, the checkpoint centers not only on the control activities assigned to the auditors, but also on determining the structural state of the network topology to decide how to restore the control (**PC-1**). An easy way to do this, would be to use Algorithm 1, which compiles a set of restoration strategies (STG) specified in [22] such as:

- Restoration without any type of constraint. In [22] it is called basic relink (**STG-1**) with a computational cost of $O(kn^2)$. Note that for the computation of the complexity, we always explore the worst case scenarios as addressed in [22], and simplify the study to $\mid V \mid \sim n$ and $\mid E \mid \sim e$.
- Restoration based on the graph diameter to minimise the intrinsic problem of the non-locality of PDS, denoted in [22] as diameter-based relink (**STG-2**) with $O(kn^2)$.
- Repair through backup instances of driver nodes, where each instance is organised inside a tree-like structure based on the concept of nice tree decomposition (**STG-3**) with an overhead of $O(k(\sum_{bk=1}^{M}(2^{w+1}(b+e))))$.

However, Algorithm 1 also includes a fourth strategy (**STG-4**) associated with the insertion of new edges or nodes, where the detection of cycles after insertion has to be addressed using the function isDAG. If there are loops, the approach has to remove them using, for example, the Berger-Shor algorithm for digraphs defined in [26] with a computational cost of $O(\mid V \mid\mid E \mid) = O(ne) \approx O(n^2)$, such that $\mid V \mid \approx \mid E \mid$ in the worst case. Note that this process might bring about numerous changes in the network topology where the number of unobserved nodes (U) can increase in proportion to the number of insertions or cycles. For this reason, the algorithm has to verify the observation degree of the entire network by verifying the completeness of **OR1** defined in [23] with a cost of $O(n^2)$. As a result, we determine that the total cost involved in the (acyclicity and completeness) tests and the removal procedure in **STG-4** is therefore: $O(n+e) + O(n^2) + O(n^2) = O(n^2)$. Nonetheless, as this process only restores **OR1**, verifying the fulfilment of **OR2** is also required as specified in [22], which adds an additional cost of $O(n^2)$. Computing all these costs, the resulting overhead for **STG-4** is hence of quadratic order. After repair, a new network structure with a new control relationship may originate in which a new selection of new auditors can be needed, considering, in this case, the two selection conditions given above, $B^{A+}_{cond1,2}$.

In order to detect structural changes, each node of the network has to periodically validate its incoming and outgoing connections so as to check for unexpected changes, probably arising from: (i) threats of type **SCN-1,2,3**; (ii) the insertion of new (not necessarily malicious) members within the network (equivalent to **SCN-3** but without malicious actions), and (iii) the leaving of legitimate nodes from a given network (equivalent to **SCN-1**). To do this, we also have to assume that each node of the network preserves, in local, a routing table with at least the information from its parents (the incoming ones or the in-degree connections, d^-) and its children (the outgoing ones or the out-degree connections, d^+) such as their IDs.

As stated by Dwork *et al.* [27] and Agbaria *et al.* [24], the network is partially synchronous. The nodes communicate with each other and carry out the control processes, but if one of them detects a structural variation within its neighbourhood, this node

ALGORITHM 1: Combined Restoration Strategies

```
Input: $\mathscr{G}(V,E)$,$\mathbf{N}_D$, $A$, $U$, $STG-Removal$;
Output: $\mathbf{N}_D$;

$or1 \leftarrow$ FALSE; $or2 \leftarrow$ FALSE;
if $(A \neq \oslash)$;
then
    if not($\mathscr{G}(V,E)$ is DAG);
    then
        comment: STG-4 includes the tests of acyclicity
        and the removal of cycles;
        $\mathscr{G}(V,E) \leftarrow$ Cycle Removal($\mathscr{G}(V,E)$);
        $U \leftarrow$ Unobserved Nodes($\mathscr{G}(V,E)$,$\mathbf{N}_D$);
        $or2 \leftarrow$ TRUE;
    end
    comment: STG-1,2,3 are specified in [22];
    while $(U \neq \oslash)$;
    do
        Randomly choose a vertex $u \in U$;
        if $(u \notin N_D)$;
        then
            if $(STG\text{-}x = 1)$;
            then
                $N_D \leftarrow$ Basic Relink($\mathscr{G}(V,E)$, $\mathbf{N}_D$,
                $U, A$);
            else
                if $(STG\text{-}x = 2)$;
                then
                    $N_D \leftarrow$ Diameter-Based Re-
                    Linked($\mathscr{G}(V,E)$, $\mathbf{N}_D$, $U$, $A$);
                else
                    $T_{bk} \leftarrow$ Backup
                    OR2($\mathscr{G}(V,E)$)$^a$;
                    $N_D \leftarrow$ Backup
                    Instance-Based
                    Scheme($\mathscr{G}(V,E)$,$N_D$,$A$,$U$,$T_{bk}$,
                    $M$);

                end
            end
        end
    end
end
if $or1$;
then
    Observation Completeness($\mathscr{G}(V,E)$,$\mathbf{N}_D$))$^b$;
end
RETURN Common verifyOR2($\mathscr{G}(V,E)$,$\mathbf{N}_D$,$A$,$or2$)$^c$;
```

[a] Backup OR2 generates the backup tree based on driver nodes.

[b] Observation Completeness corresponds to the verification process of the rudimentary **OR1** defined in [23].

[c] verifyOR2 is specified in [22].

must then alert all those nodes located at 1-hop within its neighbourhood. Therefore, we take the following protection criterion:

[**PC-2**] *Upon suspicion of a threatening situation, all nodes within the network must asynchronously cooperate with each other and under a multicast mode to alert the inspectors (internal/main auditors) as soon as possible.*

Although the propagation of the warnings is done in multicast mode, the messages have to be widely extended to the rest of nodes of the network until reaching the entire system, unless the nodes have already received the same message (duplications have to be avoided). This collaboration not only ensures that this information reaches the auditors (**PC-2**), but also guarantees the checkpoint in these nodes. For the checkpoint, the auditors may require the handling of additional information from the network to determine its current state and its control level, storing this information inside a local memory for future validations. Apart from this, it is largely assumed that all the control devices are able to store past evidence what may help the decision-making processes to identify threat scenarios and criticality levels. For example, a threat of type **SCN-2** or **SCN-3** (in which the ID of the observed node has not passed the authentication mechanisms) might be more aggressive in critical infrastructures than a threat of class **SCN-1**. However, this last criterion depends on the security policies of each organisation and on the criticality level that a threat could trigger within a specific CPS.

For a hierarchical multicast communication, each node $\in V$ has to perform an aggregation process before sending this information to the rest of the neighbours as illustrated in [28, 29]. This aggregation permits a history of the graph to be obtained, holding all those IDs that have led to an unforeseen crash or misbehaviour in the past. Therefore, we define the third protection condition as follows:

[**PC-3**] *Intermediary nodes must "aggregate" in their systems (i.e. in local) the list of IDs related to those suspicious nodes included within the warning message received from one or several parent nodes, and cooperate in the propagation of the alert by forwarding such information to the rest of neighbours located at one-hop* (**PC-2**).

4.1 Sets for the Protection, Agreement Protocol and Diagnosis

In order to address **PC-1,2,3**, the nodes and their respective auditors must manage and update the following sets of data (see Fig. 2):

- ***R***: This set contains routing information of at least those nodes located at 1-hop, composed of the IDs belonging to parents and children (the d^+ and d^-).
- ***J***: It holds those new members with recognised identities given by the security policies or by the organisation itself.
- ***L***: It contains those legitimate nodes that decide to leave the network at a given moment. However, there also exists the case where control nodes are not properly recognised by the network or their inspectors, and they need to be isolated or expelled from the network. To make a distinction between legitimate and unrecognised nodes, we use set L^* to represent any expulsion from the network, such that $L \subseteq L^*$.

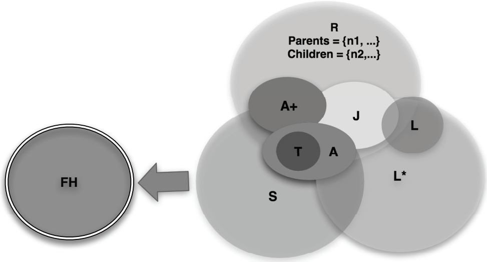

Figure 2: Sets for fault-intrusion detection model

- ***S***: This set comprises all those suspicious nodes that have more than likely triggered a fault within the network. As its initial indicates, these nodes are under "suspicion", so that auditors must investigate and determine the degree of goodness of the suspicious node, possibly with the help of the entire network.

- ***T***: When a node or an auditor detects that a node in the network is misbehaving in the structural controllability, it is included within T (the set of the possible threats $-$ which can contain both compromised nodes and/or faulty nodes). That is, this set encompasses those "recognised or unrecognised" nodes which have led to, for example: An unexpected insertion of edges from an unrecognised ID to a determined node of the network (**SCN-3**); or several attempts to cause (**SCN-1/2**), which could require an *agreement protocol* between auditors so as to find a way to isolate the problem immediately.

- ***FH***: A record composed of past evidence based on IDs. This set simply offers a record of activities.

As these sets have to be periodically monitored and updated, the auditors have to drive a sequence of validation stages so as to diagnose the occurrence of a fault caused by a node $\in \mathscr{G}$. This procedure, similar to that considered by Khanna *et al.* in [29], is as follows:

Step 1 (General): If a node v_i of the network detects a structural change, it has to comply with **PC-1,2,3**. Namely, first of all, v_i has to frequently look at R to check the current connections with respect to the connections defined within R. If some connections (either parents or children) are not available but are defined in R, it means that v_i has possibly received a threat of type **SCN-1**. In these circumstances, v_i updates its set $S = \{v_k, \ldots, v_j\}$ such that $(v_i, v_k) \in V, \ldots, (v_j, v_i) \in V$ (both parents and children). In the rare case where v_i detects that one of its parents is not part of R, it also then determines that it has probably been targeted by some new suspicious control link (**SCN-3**). In view of this, v_i updates T with the ID of the suspicious node (e.g. node 6 where $\mathrm{ID}_i = 11$ in Fig. 3). Note that it can also be the case that new nodes J want to join the network. In this case, the new nodes must establish a welcome protocol using, for example, a joining request holding a HELLO (see Fig. 3). During this procedure,

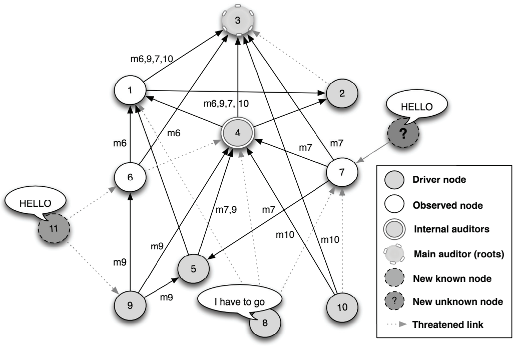

Figure 3: Communication and warning

legitimate nodes have to decide whether or not to refuse the communication when they receive a request of HELLO from unrecognised IDs (e.g. with an ID$_i$ from a reserved threshold of identities), in addition to notifying of the situation by updating T. Note that this welcome process in real contexts should be subject to restrictive and strong user authentication and integrity mechanisms so as to detect, for example, impersonations or identity falsification.

The opposite scenario is when existing nodes that want to leave the network. In this case, they must notify their neighbours, located at one-hop, of the situation so that their set L can be updated (see Fig. 3). If L and/or J are updated, then S should not be upgraded. Nonetheless and despite not suspecting a threat/fault, it should be obligatory to alert the rest of the network to any structural change using, for example, a variable of "*change* = true". Through this variable it is possible to know whether (or not) a restoration process of the controllability is needed at a given moment.

Once the threat has been detected, v_i must alert the rest of nodes located at 1-hop, and propagate the warning m_{v_i} to the entire network, thanks in part to the collaboration of the whole graph (see Fig. 3). However, there exists the case where a node cannot propagate the warning since all its children or the node itself might have been affected by a fault, partially isolating it (e.g. node 2 in Fig. 3). To detect this situation, the auditors must periodically and forcibly launch a diagnostic process of the current connections of the network (go to Step 2).

Step 2 (Auditors): When an auditor $a_i \in A^+$ receives a valid m_{v_i} containing "*change* = true", and/or $0 <\mid S \mid$ or $0 <\mid T \mid$, it first needs to clarify the situation (through a checkpoint) before proceeding with the repair of the controllability. This procedure has several stages. First, $a_i \in A^+$ verifies whether T (which contains the most critical threats previously known by the nodes – e.g of type **SCN-3**) is not empty. If T is not empty, the auditor should then confirm, for each ID contained within T, whether they are truly reachable from the auditor at a determined time $t \leq \tau_{a_i,v_i} + \Delta t$ such that τ_{a_i,v_i} represents the number of hops between the auditor and the diagnosed node v_i, and Δt shows the estimated time delay [29].

We compute the number of hops using the minimum diameter as stated in [30] such that $\Delta t = 1ms$[1]. As the network distribution follows a power law, the diameter becomes $d \sim Lnlnn$ such that $n \sim \mid V \mid$, whose value is even smaller than small world networks $O(lnn)$ and remains almost constant during the growth of the network [31, 32]. To compare distances (past-present), we need to pre-compute the minimum diameter for each auditor, such that $\forall v_k \in V, \tau_{ai,v_k} + \Delta t$. This information is stored in an array-like structure to help auditors compare the distances between peers at execution time.

If the node $v_j \in T$ has a recognised ID assigned to it and is easily reachable from the auditor and within a "known" time $t \leq t \leq \tau_{a_i,v_j} + \Delta t$, then the auditor concludes that the source node v_i, which released the initial warning, made a serious mistake on this occasion. As the source node v_i has made a mistake, both T and FH (belonging to the auditor) have to be updated by removing v_j from T and including v_i as part of FH.

After reviewing T, the auditor must then validate the identifiers contained in S. The procedure is similar to the one described above. For each node included in S, the auditor verifies the distance between the auditor and the ID, contrasting the reach time t with the pre-stored information $\tau_{a_i,v_j} + \Delta t$, such that $1 \leq t \leq \tau_{a_i,v_j} + \Delta t$. If the node v_j is not reachable in the expected time, the auditor needs to confirm the type of fault, either **SCN-1** or **SCN-2**. As we consider that a fault of type **SCN-2** may become graver than a threat of **SCN-1**, it is necessary to verify this threat before **SCN-1**. To do so, the auditor forwards the warning to the main auditor/s (located in its reach and according to its routing table) and in unicast mode; after this go to Step 3. If the main auditor/s cannot reach the observed node, then they confirm the possible threat of class **SCN-2** to the auditor demanding this information. Once the threat is confirmed, this auditor declares that it is necessary to repair the control, but this time isolating the suspicious node from the network (updating L^* and FH). In this way, it is possible to protect the control while the central system and/or human operators are being alerted to the situation. It is also possible to be in the situation where the auditor a_i either is unable to reach its main auditors, or it is in fact one of the main auditors. In either of these two scenarios, the auditor determines the scope of the problem and proceeds to the restoration taking into account $B^{A+}_{cond1,2}$.

If one of the main auditors is able to reach the suspicious node, then it confirms the existence of a fault of type **SCN-1** and updates FH. In addition, the auditor also has to compute the times (the frequency) that v_j has led to a fault in the past, using in this case, the set FH. On the other hand, as our model is based on a collaborative system for fault management, the problem can require a minimum of $\mid V \mid \geq 3f + 1$ faults [31, 29, 24]. Hence, the number of faults is bounded to f.

As stated in Step 1, intermediary nodes between a_i and v_i can also isolate themselves to interrupt the propagation of a possible warning, m_{v_i}, or be attacked. In order to detect both situations, the auditors have to review the current state of the network by observing the sources and intermediary nodes implied in the propagation of a warning (i.e the "from or across"). If one node $v_j \in \mathscr{G}$ (or several) is not reachable by a_i, then the auditor demands help from its main auditor/s to determine (where there is) a possible isolation of v_j (i.e. **SCN-2** or **SCN-1**).

[1] In real contexts, we should consider not only the time required for the query, but also the implicit time for its ACK, with a double value for t.

ALGORITHM 2: Nodes

```
Input: $\mathscr{G}(V,E)$, $J$, $L$;

$stop \leftarrow$ FALSE;
repeat
    for ($t \in Clock^{a}$);
    do
        $change \leftarrow$ FALSE; $T \leftarrow \oslash$; $S \leftarrow \oslash$;
        comment: Verify the existence of new members;
        for each ($v_j \in J$);
        do
            if ($v_j \in [ID_x...ID_y]^{b}$);
            then
                if ($v_j \in L$) and ($v_j \notin S$);
                then
                    comment: This means that $v_j$ was part of the network in the past;
                    $L \leftarrow L \setminus \{v_j\}$;
                end
            else
                comment: Possible suspicion of threat of type SCN-3;
                $T \leftarrow T \setminus \{v_j\}$;
            end
            $change \leftarrow$ TRUE;
        end
        comment: Examine the around a node.;
        if ($L \neq \oslash$);
        then
            $change \leftarrow$ TRUE;
        end
        $dif \leftarrow (R_{parents}{}^{c} - d^{-}_{current}) - L$;
        if ($dif \neq \oslash$);
        then
            $S \leftarrow S \cup dif$; $change \leftarrow$ TRUE;
        end
        $dif \leftarrow (R_{children}{}^{d} - d^{+}_{current}) - L$;
        if ($dif \neq \oslash$);
        then
            $S \leftarrow S \cup dif$; $change \leftarrow$ TRUE;
        end
        if change;
        then
            $R \leftarrow$ Update Routing Table($\mathscr{G}(V,E)$,$v_i$);
        end
        if ($S \neq \oslash$) or ($T \neq \oslash$) or change;
        then
            Send Warning(ID$_{so}{}^{e}$);
        end
    end
until (not $stop$);
```

[a] The node checks its surrounding connections periodically.

[b] $[ID_x...ID_y]$ corresponds to a range of recognised identities for authentication.

[c] $R_{parents}$ obtains the parent connections from R.

[d] $R_{parents}$ obtains the children connections from R.

[e] ID$_{so}$ represents the ID of the source node that sent the warning.,T, S, L, *change*.

Step 3 (The main auditors): The main auditors (the roots) perform the same tasks as an internal auditor (in Step 2). However, they can also sometimes be requested by other auditors to check the scope of a particular node due to their overall vision of the graph.

Both Algorithm 2 and Algorithm 5 outline the semantic description of the fault and intrusion detection model. Their correctness proofs are solved when the following requirements are satisfied: (i) The algorithm that restores, ensures controllability without violating the structural control properties (*restoration*); the algorithm is able to properly detect changes in the structural controllability and address the recovery process (*termination*); and the algorithm is able to automatically restore the monitoring network and provide dynamic control at any moment (*validity*). For the former requirement, we particularly look to Algorithm 1 where the observation degree of the entire network (**OR1**) is always considered after recovery (the validity of **STG-1,2,3** as specified in [22]), and the verification process of **OR2** is always carried out to guarantee the power of the dominance.

Through induction we show the termination of the restoration and the continuity of the network, where we first define the initial and final conditions, and the base cases. Namely, for Algorithm 2:

Precondition: There exists a structural change due to $J \neq \oslash$, $L \neq \oslash$ or the existence of a possible threat of type **SCN-1,2,3,4** at a time t.

Post-condition: Automatic restoration of **OR1** and **OR2**.

Case 1: $J \neq \oslash$ such that $\mid J \mid \geq 1$. This means that each node v_i with relationship to a $v_j \in J$ has to validate the authenticity of ID_{v_j}. When doing this, three situations can arise:

- v_j has an invalid identification associated with it, so it is considered a threat of class **SCN-3**. In this situation, v_i updates its set T and propagates the warning to the rest of the network in order to expel v_j through Algorithm 5.
- The ID of v_j is recognised by the system and is not part of sets S and L. Thus v_i has to update its routing table.
- The ID of v_j is recognised and the node is not part of set S but it is indeed included in L. This means that v_j corresponded to the network, at least once, in the past. The process finishes updating L and the routing table.

Case 2: $L \neq \oslash$ such that $\mid L \mid \geq 1$. Any node v_i with a relationship with one node of L has to update its R and indicate to the auditors the need to repair the control through the variable *change* (= true).

Case 3: Suspicion of threat **SCN-1,2,3** or **SCN-4**:

- **SCN-3**: v_i detects that one (or several) of the incoming links is not included in R. This signifies that there has been an intentional insertion of a link, and the node v_i has to update its T, leaving the identity verification process to the auditor.
- **SCN-1,2**: v_i observes that one (several $-$ **SCN-1** $-$, or all $-$ **SCN-2**) of the incoming links and/or outbound links belonging to its set R, is not enabled. This forces the update of S.

- **SCN-4**: The sum of conditions given for **SCN-3** and **SCN-1,2**.

ALGORITHM 3: SCN-1,2

```
Input: $\mathscr{G}(V,E)$, $\mathbf{N}_D$, $S$, $FH$, $L^*$, $a_i$, $diameterV$, $threat$,
    $restore$;
Output: $FH$, $threat$, $restore$;

if ($S \neq \emptyset$);
then
    for each ($v_s \in S$);
    do
        $d \leftarrow$ BFS($\mathscr{G}(V,E)$, $a_i$, $v_k$);
        if ($a_i \in A^+ \neq v_s$) and ($diameterV[v_s] \neq d$) and
        ($v_s \notin L^*$);
        then
            comment: Forwarding to the main auditors;
            $isolation \leftarrow$ SEND TO THE MAIN AUDITORS($a_i, v_s$);
            if ($isolation \leftarrow TRUE^a$);
            then
                comment: Possible threat of the class
                SCN-2.;
                $FH \leftarrow FH \cup \{v_s\}$;
                $threat \leftarrow threat \cup \{v_s\}$;
                $restore \leftarrow$ TRUE;
            else
                $FH \leftarrow FH \cup \{v_s\}$; $cont = 0$;
                for each ($v_{fh} \in FH$);
                do
                    if ($v_{fh} = v_s$);
                    then
                        $cont \leftarrow cont + 1$;
                    end
                end
                if ($cont > ((|V| - |L^*|) - 1)/3$);
                then
                    comment: SCN-1, abuse of
                    faults or the node is faulty;
                    $threat \leftarrow threat \cup \{v_s\}$;
                    $restore \leftarrow$ TRUE;
                end
            end
        else
            $FH \leftarrow FH \cup \{\mathrm{ID}_{so}\}$;
        end
    end
end
RETURN $FH$, $threat$, $restore$;
```

[a] This means that no main auditor is able to reach v_s.

Note that any structural change (e.g. new links or the removal of edges) or suspicion of threat (either by an attack or a fault) entails a warning to the network.

Induction: In stage k of the repeat (with $k > 1$) with $t > 0$, Algorithm 2 has to periodically verify three possible structural alterations, complying with **PC-1** and **PC-2**:

- Insertion of new links, likely due to the incorporation of new members. In this case, **Case 1** of this proof must be launched.

- Removal of edges because existing members are going to leave the network. These nodes have to advise they are leaving the network through L (see **Case 2**).
- Unexpected addition/deletion of links within the network, where the nodes have to update S or T according to the current state of their d^{+} and d^{-} (see **Case 3**).

In all these cases, the variable *change* must be updated to indicate the need to restore the structural controllability. If such a need does indeed exist, a warning message is created and distributed to the rest of the network (**PC-2** and **PC-3**). This process is periodically repeated at each time t of the clock, and Algorithm 2 never finishes because the node is always active. However, the post-condition is true when the post-condition of Algorithm 5 is also true.

ALGORITHM 4: SCN-3

```
Input: $\mathscr{G}(V,E)$, $\mathbf{N}_D$, $T$, $S$, $FH$, $L^*$, $a_i$, $diameterV$, $threat$,
       $restore$;
Output: $FH$, $S$, $threat$, $restore$;

if ($T \neq \oslash$);
then
    for each ($v_t \in T$);
    do
        $d \leftarrow$ BFS($\mathscr{G}(V,E)$, $a_i$, $v_t$);
    end
    if ($a_i \in A^+ \neq v_t$) and ($v_t \in [ID_x...ID_y]$) and
    ($diameterV[v_t] = d$) and ($v_t \notin L^*$);
    then
        $FH \leftarrow FH \cup \{\mathrm{ID}_{so}\}$;
    else
        if ($v_t \notin [ID_x...ID_y]$);
        then
            $threat \leftarrow threat \cup \{v_t\}$;
        else
            $S \leftarrow S \cup \{v_t\}$;
            $FH \leftarrow FH \cup \{v_t\}$;
            $restore \leftarrow$ TRUE;
        end
    end
end
RETURN $FH$, $S$, $threat$, $restore$;
```

As indicated in the proof of Algorithm 2, the post-condition becomes true when the post-condition of Algorithm 5 is also true. The proof of Algorithm 5 is given as follows:

Precondition: As defined in post-condition of Algorithm 2, there exists a suspicion of structural change because $L \neq \oslash$, $S \neq \oslash$ or $T \neq \oslash$ at a given moment t.

Post-condition: Restored structural controllability, confirming the post-condition of Algorithm 2.

Case 1: $T \neq \oslash$ such that $\mid T \mid \geq 1$. In this case, the auditor has to validate the identity of the node $v_j \in T$ (or each node holding in T) and its reach to confirm its accessibility within the network. Under these conditions, several situations can arise:

- If v_j in T is recognised but is not reachable at $1 \leq t \leq \tau_{a_i,v_j} + \Delta t$, the auditor determines that possibly there is a threat of type **SCN-1,2**. In order to verify this hypothesis, the auditor includes the node as part of S (the suspicious ones – **Case 2** of this proof).

ALGORITHM 5: Auditor a_i

```
Input: $\mathscr{G}(V,E)$, $\mathbf{N}_D$, $STG - Removal$, $L^*$;

$restore \leftarrow$ FALSE; $change \leftarrow$ FALSE; $T \leftarrow \emptyset$; $S \leftarrow \emptyset$; $L \leftarrow \emptyset$; $J \leftarrow \emptyset$; $threat \leftarrow \emptyset$;
$diameterV$[a] $\leftarrow$ BFS[b]($\mathscr{G}(V,E)$);
repeat
    RECEIVE WARNING(ID$_{so}$, Inter[c],$T$, $S$, $L$, $change$);
    comment: Check the state of the entire network;
    for each $(v_j \in \mathscr{G}(V,E))$;
    do
        $d \leftarrow$ BFS($\mathscr{G}(V,E)$, $a_i$, $v_j$);
        if $(a_i \in A^+ \neq v_j)$ and $(diameterV[v_j] \neq d)$ and $(v_j \neq ID_{so})$ and $(v_j \notin Inter)$ and $(v_j \notin L^*)$;
        then
            $FH \leftarrow FH \cup \{v_j\}$;
            $change \leftarrow$ TRUE;
        end
    end
    if $(change = TRUE)$ or $(T \neq \emptyset)$ then
        comment: Possible threat of type SCN-3 − Algorithm 4;
        $\{FH, S, threat, restore\} \leftarrow$ SCN-3($\mathscr{G}(V,E)$, $\mathbf{N}_D$, $T$, $S$, $FH$, $L^*$, $a_i$, $diameterV$, $threat$, $restore$);
        comment: Threat of type SCN-1 or SCN-2 − Algorithm 3;
        $\{FH, threat, restore\} \leftarrow$ SCN-1,2($\mathscr{G}(V,E)$, $\mathbf{N}_D$, $S$, $FH$, $L^*$, $a_i$, $diameterV$, $threat$, $restore$);
        if $(L \neq \emptyset)$ and $(L \nsubseteq S)$;
        then
            $restore \leftarrow$ TRUE;
        end
    end
until $(threat \neq \emptyset)$ or $restore$;
$L^* \leftarrow L^* \cup (threat \cup L)$;
$U \leftarrow$ UNOBSERVED NODES($\mathscr{G}(V,E)$),$N_D$;
COMBINED RESILIENCE($\mathscr{G}(V,E)$,$\mathbf{N}_D$, $threat$, $U$, $L^*$, $FH$, $STG - Removal$);
NEW ASSIGNATION OF AUDITORS($\mathscr{G}(V,E)$, $FH$, $L*$);
if $(threat \neq \emptyset)$;
then
    ALERT THE CENTRAL SYSTEM($\mathscr{G}(V,E)$, $L^*$, $T$, $S$);
end
```

[a] $diameterV$ represents a predefined table storing the initial diameters between a_i and the rest of nodes $v_j \in \mathscr{G}(V,E)$.

[b] BFS computes the diameter through Breadth-First Search (BFS) algorithm with complexity $O(n+e) = O(n)$.

[c] Inter comprises the set of intermediary nodes $\{ID_k, \ldots ID_z\}$.

- If the node v_j is not recognised or does not pass the authentication mechanisms, the auditor confirms the existence of a threat **SCN-3** (as detected by the node v_i in Algorithm 2).
- If v_j in T is recognised and is reachable at $1 \leq t \leq \tau_{a_i,v_j} + \Delta t$, the auditor determines that possibly the sender made a serious accusation. So the auditor a_i penalises the sender by including it in FH.

In any of these cases, FH has to be updated for the future, and only in those cases where there does indeed exist a structural change, is the variable *restore* updated to facilitate the exit from the *repeat* instruction.

Case 2: $S \neq \oslash$ such that $\mid S \mid \geq 1$. In this case, the auditor has to validate the accessibility of the node v_j contained in S (or for each node in S) and from the auditor. As for **Case 1**, several scenarios may be seen:

- The node v_j is not reachable at $1 \leq t \leq \tau_{a_i,v_j} + \Delta t$, so the auditor determines the possibility of a threat of type **SCN-2**. To prove this supposition, the auditor forwards the warning to the root (or reachable roots from the auditor − Step 3 defined above) of the system so as to check its complete isolation. Depending on the roots, we could encounter a further two situations:
  - There is indeed an isolation in v_j. Given this, the auditor confirms its suspicion and considers the node v_j in S as a threat.
  - No isolation exists, so the auditor updates FH and verifies the number of times that the node v_j has made an error/fault in the past (i.e. the number of times/occurrences that an ID_j appears in FH). Given that we have bounded the number of faults to f such that $\mid V \mid \geq 3f + 1$ [31, 24, 29], if the auditor observes an excess of faults (greater than f), it then determines a threat − either due to a targeted attack or a faulty node.
- The node v_j in S is reachable from the auditor a_i at a time $1 \leq t \leq \tau_{a_i,v_j} + \Delta t$; hence, this obliges a_i to update FH using the identity of the sender.

In either case, FH is always updated for the future, in addition to examining the state of the variable *restore*.

Case 3: $L \neq \oslash$ such that $L \nsubseteq S$. As in **Case 2** of Algorithm 2, this base case restores the structural controllability since a (legitimate) node leaving $\mathscr{G}$ has addressed the total elimination of its edges (similar to **SCN-1**). To do this, *restore* has to be updated to validate the condition of the *repeat*.

Induction: In step k of the *repeat* (with $k > 1$) with $t > 0$, Algorithm 5 has to validate the following conditions:

- A threat of type **SCN-3** through **Case-1** of this proof.
- A threat of class **SCN-1,2** or the existence of faulty nodes by limiting the number of faults to f (see **Case 2**).
- Leaving such that $L \neq \oslash$ (corresponds to **Case 3**).

If one of these four conditions entails a new change in the control of the network, the *repeat* instruction is stopped by **PC-1**. Once the loop has been stopped, Algorithm 5 has to (i) isolate those nodes that are suspected of being a threat or those nodes that want to leave the network; (ii) obtain the observation rate by verifying **OR1**; and (iii) repair the structural controllability using Algorithm 1. It is important to highlight at this point that the restoration process can further restrict the search for those candidates involved in the repair procedure. In a nutshell, the search for new driver nodes through **OR1** and **OR2** [23] should also be restricted to FH. If a driver node has recently caused several faults, this node should not be selected (as far as this is possible) for the recovery of the controllability (i.e. for the new N_D).

On the other hand, as the scenario is dynamic with the possibility of having to adapt mobile technology (e.g. smart phones, tablets, etc.), the network topology and its control structure can vary significantly, so it is essential not only to validate the state of **OR1** but also that of **OR2**. Obviously, this verification procedure can involve new topologies and structures within the control hierarchy where a new set of auditors should be selected in order to respect the structural properties of the controllability and the degree sequence of the power-law distribution ($B^{A+}_{cond1,2}$). This is done through the *new assignation of auditors* function which must consider the state of FH, but this time looking at the behaviour of the auditors; or rather, looking at their misbehaviour by quantifying their faults. In this way, it is possible to further restrict the selection of candidates according to the misbehaviour of auditors in the past.

In the case of a threat, the central system or human operators should be alerted to the situation, providing them with the maximum amount of information, needed to attend to the situation, such as the type of threat detected, the isolated nodes (identification, location, etc.) and the sets of T and S. After this, the post-condition is true and Algorithm 5 concludes.

With this validation, the last requirement (the *validity*) described above, is also satisfied since Algorithm 5 is able to detect disturbances in the structural controllability (verified through the induction) and ensure the continuity and resilience without infringing the two observation rules (through Algorithm 1). With respect to the complexity analysis, the cost invested by the approach depends on the implicit complexity of **STG-1,2,3** as analysed in [22], but extending the study to consider the **SCN-4** scenario. In other words, assuming that two nodes within $\mathscr{G}$ can be attacked such that $a_1, a_2 \in A$ (the set of the attacked), the permutation of threat scenarios can look like: $a_1, a_2 \in$ **SCN-1**; $a_1 \in$ **SCN-1** - $a_2 \in$ **SCN-2**; $a_1 \in$ **SCN-1** - $a_2 \in$ **SCN-3**; $a_1 \in$ **SCN-2** - $a_2 \in$ **SCN-2**; $a_1 \in$ **SCN-2** - $a_2 \in$ **SCN-3**; and $a_1 \in$ **SCN-3** - $a_2 \in$ **SCN-3**. The computational cost invested in these combinations is shown in Table 1. From this table, we observe that **STG-3** combined with **STG-4** is the least suitable choice, its complexity tends to an exponential order in the worst case. Conversely, **STG-1** and **STG-2** combined with **STG-4** seems to be the best option, following a quadratic order in all cases. At this point, it is also important to underline the current technical benefits of the majority of control devices for local storage of large evidence streams based on lightweight values (e.g. the ID of devices, date). For example, most industrial control sensors such as ISA100.11a or WirelessHART work at $\sim$ 4MHz-32MHz micro-processor, 8KB-128KB RAM, 128KB-192KB ROM centralizing the evidence in powerful servers or gateways [33]; whereas RTUs generally work at $\sim$ 22MHz-200MHz with 256 bytes-

Table 1: Complexity for the Four Combined Restoration Strategies

Strategies	SCN-$_x$ - SCN-$_x$	Complexity	Dimensionality
STG-1,2 - 4	**SCN-1 - SCN-1**	$O(kn^2)$	n_d+4
	SCN-1 - SCN-2	$O(kn^2)$	n_d+3
	SCN-1 - SCN-3	$O(kn^2)$	n_d+3
	SCN-2 - SCN-2	$O(kn^2)$	n_d+2
	SCN-2 - SCN-3	$O(kn^2)$	n_d+2
	SCN-3 - SCN-3	$O(n^2)$	n_d+2
STG-3 - 4	**SCN-1 - SCN-1**	$O(k(\sum_{bk=1}^{M}(2^{w+1}(b+e))))$	n_d+4
	SCN-1 - SCN-2	$O(k(\sum_{bk=1}^{M}(2^{w+1}(b+e))))$	n_d+3
	SCN-1 - SCN-3	$O(k(\sum_{bk=1}^{M}(2^{w+1}(b+e))))$	n_d+3

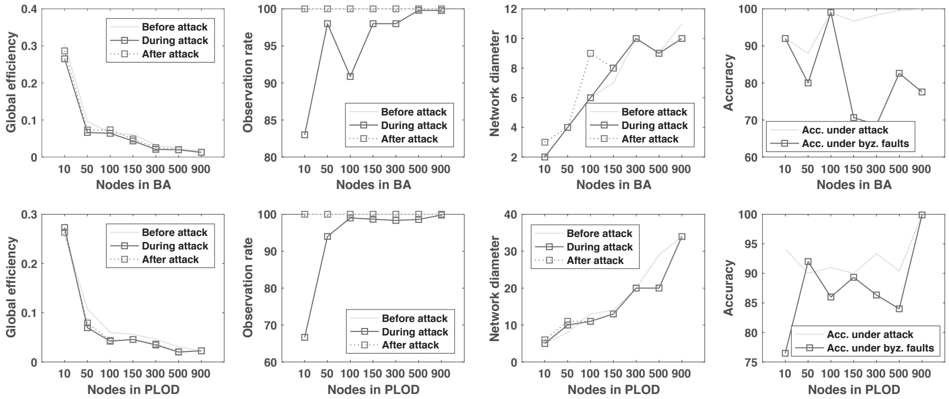

Figure 4: Degradation and reparation in BA and PLOD distributions

1GB RAM, 8KB- 32MB flash memory and 16KB-256KB EEPROM [33] with the capacity to incorporate an external hard drive such as [34] or rely on external backup infrastructures, such as private cloud or fog-computing systems [35, 36] – note that this last focus corresponds to one of our future work and as an extension to our previous work [36].

As for the dimensionality of N_D, we observe that the best threat scenarios that avoid abrupt changes in the cardinality of the set of driver nodes are those related to **SCN-2**-**SCN-2** (isolation), **SCN-3**-**SCN-3** (injection of control) and **SCN-2**-**SCN-3**. In contrast, scenarios of type **SCN-1** (i.e. **SCN-1** and **SCN-1**) are the worst threat models for N_D, meaning that an arbitrary removal of edges may become much more devastating for the dimensionality of N_D than a threat in the isolation of nodes. Still, this functional characteristic largely depends on the number of nodes that infringe **OR1** and **OR2**; a feature that is also stated in [23]. Lastly, and regarding communication in **SCN-4** scenarios, we have already discussed in Section 4.1 that the diameter in power law networks becomes $d \sim Lnlnn$ such that $n \sim \mid V \mid$, the value of which is smaller than the diameter of small world networks ($O(lnn)$), remaining almost constant during the network's growth. This feature means that the communication overhead in power-law distributions leads to $O(Lnln \mid V \mid)$ messages, if and only if, there exists a multicast communication without duplications as specified in [31]. Also note that within this overhead for its practical use in real contexts, it is also essential to consider some other factors such as the multiple network interconnections and protocols [37], which can certainly affect certain network parameters increasing delays or collisions. However, many industrial network protocols are designed to consider part of these problems providing coexistence techniques such as frequency agility/hopping or blacklisting methods [38].

4.2 Practical Validation and Results

This section shows a practical case study carried out in PLOD and BA distributions with a small connectivity probability to represent realistic scenarios, such as $\alpha = 0.2$ and $\alpha = 3$, respectively. To do this, we simulate several scenarios in which the communication links are slightly perturbed so as to produce substantial changes within the network where: (i) the number of threats is restricted to $\delta \leq \frac{|V|}{2}$ and (ii) there exists a high possibility of new insertions of nodes or the removing of nodes from the network (as stated in Table 2). The model stops when an auditor (the winner) detects a known threat (**SCN-1,2,3,4**) or the existence of a faulty node.

Concretely, Figure 4 represents the rates of global efficiency, the rate of observation and the network diameter to illustrate the degradation of the network after an unforeseen variation or a threat, which can be originated by an intentional attack or a causal fault. In this respect, the global efficiency is computed, taking into account the average efficiency of a specific node over all the nodes in the network, such that:

$$E_g = \frac{1}{|V|(|V|-1)} \sum_{v_i, v_j \in V, v_i \neq v_j} \frac{1}{d_{v_i,v_j}} \tag{2}$$

where d_{v_i,v_j} corresponds to the shortest path length between v_i and v_j in V. This indicator can become useful to observe the degradation of the network when new or existing links are altered, either because there is a special mobility within the network or because there are certain anomalies that should be considered. This feature is also illustrated in Figure 4 for both distributions in which the global efficiency is degraded after the multiple topological changes; an effect that is also shown with the network diameter and the rate of observation.

The network diameter is computed taking into account the BFS method to get the minimum diameter between two nodes, whereas the rate of observation is based on the computation of **OR1** and in the way of extracting the set of unobserved nodes from $\mathscr{G}$, i.e. the set U. In this sense, we observe that BA distributions present a rate of observation distortion slightly higher than the PLOD distributions probably due to the strong restriction of pure power-law of the PLOD structures [39]. At this point, the resilience is reached once that the reparation mechanism is launched, which is based on the diameter-based relink strategy (**STG-2**) since its computational cost is less than the rest, as described in [22] and Table 1, with an overhead of $O(n^2)$.

Figure 4 also includes the accuracy level of our approach and its practical validation. Accuracy is a measure that computes the rate of True Positives (TPs) and True Negatives (TNs) within a given population, such that:

$$accuracy = \frac{\sum TP + \sum TN}{\sum TP + \sum TN + \sum FP + \sum FN} \tag{3}$$

To compute the accuracy, two sets F_t and A_t are considered to hold all accidental faults and attacks registered in the entire graph $\mathscr{G}$, and subsequently calculate the rates of true positives, true negatives, False Positives (FPs) and False Negatives (FNs) required for Equation 3. These rates are essential to extract the average accuracy of each auditor and identify those auditors that are truly able to detect anomalies with high accuracy. The results indicate that the detection of threats (T) is possible in both distributions, reaching significant values. Still, the approach seems to be more effective for scale-free networks than pure power-law distributions such as PLOD, in which the detection of attacks in BA networks remains almost 100% in most cases and the detection in PLOD drops to 90% detection in most cases. This scene is contrary for casual faults where the detection remains in rates of between 70%-100% in both distributions, principally because they are unforeseen faults that do not follow specific threat patterns.

Table 2: Level of mobility or threat within the network

Netw.	Nodes	$\mid L^*(L \subseteq L^*) \mid$	$\mid J \mid$	$\mid A_t \mid$	$\mid F_t \mid$
BA	10	1	2	0	0
	50	8	0	8	8
	100	1	3	0	0
	150	25	0	28	21
	300	91	0	96	3
	500	65	0	65	24
	900	153	0	144	59
PLOD	10	1	7	2	1
	50	1	0	6	3
	100	12	0	17	6
	150	6	0	13	12
	300	24	0	40	21
	500	55	0	85	27
	900	1	1	0	0

5 Conclusions and future work

A checkpoint model based on a cooperative cyber-physical network composed of trustworthy elements (auditors) has been presented. The approach is able to manage distributed warning replicas that help produce sufficient data redundancy for fault and intrusion detection in the checkpoints. Namely, through these replicas it is possible to provide resilience to the network when the control structures can be seriously threatened by attackers or perturbed by the dynamic changes caused by the joining of new members or the leaving of nodes from the network. To virtualise the approach and obtain outcomes through diverse experiments, graph theory and control theory are applied together to characterise the application context. The results show that power-law distributions in general are quite useful for accommodating checkpoint-based approaches with a high accuracy level to detect variations in the structural controllability. But the analysis also indicates that scale-free distributions can be more efficient than a pure power-law one due to the degree of centrality of their auditors.

As regards future work, we intend to extend the approach to look at additional parameters related to the integrity of the communications and messages. In this way, we could explore new network states to strengthen the approach against stronger adversaries capable of exploiting attacks related to false data injection. To do so, specific anomaly-based detection techniques (either data-mining, machine learning, statistical, etc.) could be considered as part of the inspection process so as to detect anomalous deviations, either in a part of the network or in its entirety. As this task may imply analysing large data sets, the analysis may entail the inclusion of external infrastructures for local processing support, such as a fog-computing system. Lastly, we also intend to incorporate this work into a proof-of-concept to low scale, with the aim of studying new performance parameters and further refinement.

Acknowledgment

This work has been partially supported by the research projects PERSIST (TIN2013-41739-R) and SADCIP (RTC-2016-4847-8), both financed by the Ministerio de Economía y Competitividad, as well as by the project NECS (H2020-MSCA-ITN-2015), financed from the European Unions Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 675320. The authors would also like to thank Prof. S. Wolthusen for his valuable comments during the design of the approach.

References

[1] F. Baker and D. Meyer, "Internet protocols for the smart grid," Internet Engineering Task Force (IETF), RFC-6272, 2011.

[2] F. Pasqualetti, F. Dorfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *Automatic Control, IEEE Transactions on*, vol. 58, no. 11, pp. 2715–2729, Nov 2013.

[3] E. Union, "European Union Network and Information Security Agency (ENISA)," http://www.enisa.europa.eu, last access in July 2015, 2005.

[4] D. of Homeland Security, "Industrial Control Systems Cyber Emergency Response Team (ICS-CERT)," https://ics-cert.us-cert.gov, last access in July 2015, 2015.

[5] C. Alcaraz and S. Zeadally, "Critical control system protection in the 21st century," *IEEE Computer*, vol. 46, no. 10, pp. 74–83, 2013.

[6] M. Ruchika, "Schemes for surviving advanced persistent threats," Diss. Faculty of the Graduate School of the University at Buffalo, State University of New York, 2013.

[7] B. Sanjay, S. Sanjeev, and T. Ishita, "A detailed review of fault-tolerance techniques in distributed system," *International Journal on Internet and Distributed Computing Systems*, vol. 1, no. 1, 2012.

[8] A. B. G. Veronese, M. Correia and L. Lung, "Highly-resilient services for critical infrastructures," in *Proceedings of the Embedded Systems and Communications Security Workshop*, 2009.

[9] M.Treaster, "A survey of fault-tolerance and fault-recovery techniques in parallel systems," *ACM Computing Research Repository (CoRR*, vol. 501002, pp. 1–11, 2005.

[10] C.-T. Lin, "Structural controllability," *IEEE Transactions on Automatic Control*, vol. 19, no. 3, pp. 201–208, 1974.

[11] T. Haynes, S. M. Hedetniemi, S. T. Hedetniemi, and M. A. Henning, "Domination in graphs applied to electric power networks," *SIAM Journal on Discrete Mathematics*, vol. 15, no. 4, pp. 519–529, 2002.

[12] R. E. Kalman, "Mathematical description of linear dynamical systems," *Journal of the Society of Industrial and Applied Mathematics Control Series A*, vol. 1, pp. 152–192, 1963.

[13] W.-X. Wang, X. Ni, Y.-C. Lai, and C. G., "Optimizing controllability of complex networks by minimum structural perturbations," *Phys. Rev. E*, vol. 85, p. 026115, 2012.

[14] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabasi, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, May 2011. [Online]. Available: http://dx.doi.org/10.1038/nature10011

[15] Y. Liu and A. Barabási, "Control principles of complex networks," *CoRR*, vol. abs/1508.05384, 2015. [Online]. Available: http://arxiv.org/abs/1508.05384

[16] Y. C. B. Chan and D. R. Shachter, "Structural controllability and observability in influence diagrams," in *Proceedings of the Eighth international conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1992, pp. 25–32.

[17] C. Pu, W.-J. Pei, and A. Michaelson, "Robustness analysis of network controllability," *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 18, pp. 4420–4425, 2012.

[18] J. Kneis, D. Mölle, S. R., and P. Rossmanith, "Parameterized power domination complexity," *Information Processing Letters*, vol. 98, no. 4, pp. 145–149, 2006.

[19] G. A. Pagani and M. Aiello, "The power grid as a complex network: A survey," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 11, pp. 2688–2700, 2013.

[20] C. Palmer and J. Steffan, "Generating network topologies that obey power laws," in *Global Telecommunications Conference (GLOBECOM 2000)*, vol. 1, 2000, pp. 434–438.

[21] R. A. A. Barabasi and H. Jeong, "Scale-free characteristics of random networks: The topology of the world-wide web," *Physica A.*, vol. 272, no. 2115, pp. 173–187, 1999.

[22] C. Alcaraz and S. Wolthusen, "Recovery of structural controllability for control systems," in *Eighth IFIP WG 11.10 International Conference on Critical Infrastructure*, vol. 441. Springer, 2014, pp. 47–63.

[23] C. Alcaraz, E. E. Miciolino, and S. Wolthusen, "Structural controllability of networks for non-interactive adversarial vertex removal," in *8th International Conference on Critical Information Infrastructures Security*, vol. 8328. Springer, 2013, pp. 120–132.

[24] A. Agbaria and R. Friedman, "A replication- and checkpoint-based approach for anomaly-based intrusion detection and recovery," *2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops*, vol. 2, pp. 137–143, 2005.

[25] A. Agbaria, H. Attiya, R. Friedman, and R. Vitenberg, "Quantifying rollback propagation in distributed checkpointing," *Journal of Parallel and Distributed Computing*, vol. 64, no. 3, pp. 370–384, 2004.

[26] P. Healy and N. S. Nikolov, *Hierarchical drawing algorithms*. Handbook of Graph Drawing and Visualization, CRC Press, 2013, ch. 13, pp. 409–446.

[27] C. Dwork, N. Lynch, and L. Stockmeyer, "Consensus in the presence of partial synchrony," *Journal of ACM*, vol. 35, no. 2, pp. 288–323, Apr. 1988.

[28] C. Haowen, P. Adrian, and S. Dawn, "Secure hierarchical in-network aggregation in sensor networks," in *The 13th ACM Conference on Computer and Communications Security*, ser. CCS '06. New York, NY, USA: ACM, 2006, pp. 278–287.

[29] G. Khanna, M. Cheng, P. Varadharajan, S. Bagchi, M. Correia, and P. Verissimo, "Automated rule-based diagnosis through a distributed monitor system," *Dependable and Secure Computing*, vol. 4, no. 4, pp. 266–279, 2007.

[30] J. Acosta, P. Arjona, and L. Moldes, "The impact of time delay in the connectivity distribution of complex networks generated using the barabási-albert model," *Mexican Journal of Physical*, vol. 60, pp. 145–148, 2014.

[31] G. Weldehawaryat and S. Wolthusen, "Asynchronous binary byzantine consensus over graphs with power-law degree sequence," in *Critical Infrastructure Protection*, ser. IFIP. Springer, 2014, pp. 263–276.

[32] C. Reuven and H. Shlomo, "Scale-free networks are ultrasmall," *Phys. Rev. Lett.*, vol. 90, no. 5, 2003.

[33] C. Alcaraz, L. Cazorla, and G. Fernandez, “Context-awareness using anomaly-based detectors for smart grid domains,” in *9th International Conference on Risks and Security of Internet and Systems*, vol. 8924, Springer International Publishing. Trento: Springer International Publishing, 04/2015 2015, pp. 17–34. [Online]. Available: http://link.springer.com/chapter/10.1007\%2F978-3-319-17127-2_2$\#$

[34] Honeywell, “Rtu2020 remote terminal unit specification,” SC03-300-101 Release 101 October 2014, Version 1.0, 2014.

[35] M. B. Mollah, M. A. K. Azad, and A. Vasilakos, “Secure data sharing and searching at the edge of cloud-assisted internet of things,” *IEEE Cloud Computing*, vol. 4, no. 1, pp. 34–42, Jan 2017.

[36] C. Alcaraz and J. Lopez, “WASAM: A dynamic wide-area situational awareness model for critical domains in smart grids,” *Future Generation Computer Systems*, vol. 30, pp. 146–154, 2014.

[37] ——, “Secure interoperability in cyber-physical systems,” in *Security Solutions and Applied Cryptography in Smart Grid Communications, IGI Global, USA*. USA: IGI Global, 2017, ch. 8, pp. 137–158.

[38] ——, “A security analysis for wireless sensor mesh networks in highly critical systems,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 40, no. 4, pp. 419–428, July 2010. [Online]. Available: http://ieeexplore.ieee.org/search/srchabstract.jsp?tp=\&arnumber=5443456\&queryText\%253DC.+Alcaraz\%2526openedRefinements\%253D*\%2526searchField\%253DSearch+All\&fromGateway=true

[39] C. Alcaraz and E. Etcheves Miciolino and S. Wolthusen, “Multi-round attacks on structural controllability properties for non-complete random graphs,” in *The 16th Information Security Conference (ISC)*, In Press.