Resilient Distributed Consensus for Tree Topology

Mark Yampolskiy*
University of South Alabama
Mobile, AL

Yevgeniy Vorobeychik
Vanderbilt University
Institute for Software
Integrated Systems
Nashville, TN

Xenofon D. Koutsoukos
Vanderbilt University
Institute for Software
Integrated Systems
Nashville, TN

Peter Horvath*
Budapest University of
Technology and Economics
Budapest, Hungary

Heath J. LeBlanc
Ohio Northern University
Ada, OH

Janos Sztipanovits
Vanderbilt University
Institute for Software
Integrated Systems
Nashville, TN

ABSTRACT

Distributed consensus protocols are an important class of distributed algorithms. Recently, an Adversarial Resilient Consensus Protocol (ARC-P) has been proposed which is capable to achieve consensus despite false information provided by a limited number of malicious nodes. In order to withstand false information, this algorithm requires a mesh-like topology, so that multiple alternative information flow paths exist. However, these assumptions are not always valid. For instance, in Smart Grid, an emerging distributed CPS, the node connectivity is expected to resemble the scale free network topology. Especially closer to the end customer, in home and building area networks, the connectivity graph resembles a tree structure.

In this paper, we propose a Range-based Adversary Resilient Consensus Protocol (R.ARC-P). Three aspects distinguish R.ARC-P from its predecessor: This protocol operates on the tree topology, it distinguishes between trustworthiness of nodes in the immediate neighborhood, and it uses a valid value range in order to reduce the number of nodes considered as outliers. R.ARC-P is capable of reaching global consensus among all genuine nodes in the tree if assumptions about maximal number of malicious nodes in the neighborhood hold. In the case that this assumption is wrong, it is still possible to reach Strong Partial Consensus, i.e., consensus between leafs of at least two different parents.

*The work was performed while the author was employed at Vanderbilt University, Institute for Software Integrated Systems.

HiCoNS'14, April 15–17, 2014, Berlin, Germany.

http://dx.doi.org/10.1145/2566468.2566485 .

Categories and Subject Descriptors

[**Security and privacy**]: Systems security—*Distributed systems security*; [**Security and privacy**]: Systems security—*Information flow control*; G.2.2 [**Mathematics of Computing**]: Graph Theory—*Trees*; G.1.0 [**Mathematics of Computing**]: General—*Parallel algorithms*

General Terms

Security, Algorithms

Keywords

Resilience, distributed consensus, tree topology, smart grids

1. INTRODUCTION

Several emerging classes of applications, such as Smart Grid, Vehicular Networks, or Distributed Person Tracking with multiple surveillance cameras, require coordinated behavior of all involved CPS systems. Smart Grid should be able to coordinate behavior of Smart Appliances to distribute load requirements over time and thus avoid energy spikes. Vehicular Networks should be able to distribute hazard information, thus preventing collisions and congestions on the roads. Multiple cameras can be used for coordinated distributed surveillance allowing tracking of persons in the field with the necessary grade of detail.

These are but a few of many emerging distributed CPS applications. What is common between all these applications is the need to coordinate actions of all involved CPS in a timely manner. Furthermore, often even coordination and/or optimization of local (i.e., in the neighborhood) parameters is sufficient to ensure global properties. Due to the vast amount of nodes in such scenarios as well as to the required robustness against node failures, distributed solutions should be applied.

Apart from coordinated tasks, the fundamental difference between those scenarios is that the coordination should be performed in various topologies, both network and overlay, which also have various grades of dynamicity. Smart Grids are expected to resemble scale-free networks with mesh-like core and tree-like topologies closer to the end customers,

e.g., in the Home and Building Area Networks. With exception of the Smart Appliances, which can join and leave the network comparatively often, Smart Grid network topology evolves very slowly in time; therefore, it can be considered as static. In Vehicular Networks, a highly dynamic mesh topology should be taken into account. The Distributed Person Tracking, as it has been considered in [16], operates on a static mesh. Therefore, all of these scenarios might require different solutions optimized for the particular topology and the topological dynamics.

In this paper, we focus on the consensus problem in Building Area Networks (BAN) in Smart Grid, whose network and overlay topologies are trees. As Smart Appliances, which are leaf nodes in these trees, are owned by the end customers as well as exposed to the unrestricted physical access, it is reasonable to assume that they can be rigged. Therefore, a consensus protocol resilient to the false information provided by leaf nodes is needed.

We propose a Range-based Adversary Resilient Consensus Protocol (R.ARC-P). Compared to the ARC-P protocol introduced in [7, 9], it has three main differences. First, whereas ARC-P requires mesh topology in order to reach consensus, R.ARC-P operates on the tree topology. Second, ARC-P is an indiscriminate protocol, treating all nodes in the neighborhood equally; R.ARC-P is a discriminating protocol introducing different trust relationships based on the child-parent relation in the tree topology. Third, in ARC-P, the only value always trusted is the node's own value; in R.ARC-P, the range of values between its own value and the value of the parent node is considered as trustworthy.

The remainder of the paper is structured as follows. After discussing related work in Section 2, we describe the problem formulation in Section 3. In Section 4 we present the R.ARC-P protocol. The simulation results are presented in Section 5. We discuss the reasons behind consensus with R.ARC-P as well as outline concepts of the R.ARC-P based detection and localization of malicious nodes in Section 6.2. We conclude this paper with a short review and discussion of planned future work.

2. RELATED WORK

Two areas are relevant for this paper, Smart Grid and Consensus Protocols. As our proposal is based on the ARC-P protocol, we present it in more details.

The Smart Grid is an emerging distributed CPS, which should revolutionize the way in which energy is produced, distributed, stored, and used. It should provide benefits to all stakeholders. Customers should be able to benefit from the lower energy prices as well as from the capability of Smart Appliances (SA) to adjust dynamically to daily price fluctuations. Providers can benefit because they can buy and resell electricity produced at the customer site; moreover, providers have the additional option to store energy at the customer site. Coordinated energy usage by multiple SA can reduce the energy spikes in the electric power grid, thus reducing requirements and related costs of the power distribution systems. All this should support sustainable growth of the industry and reduce impact on the ecology.

In order to accomplish these tasks, beyond the physical grid infrastructure, an extensive logical connection and coordination between all involved Smart Grid components is needed, e.g., between multiple Smart Appliances (SA) installed at the customer site and Smart Metering Infrastructure (SMI) owned by the electrical service provider.

Recent surveys of communication/networking and of routing protocols in Smart Grid are given in [3] and [15], respectively. An overview of Smart Grid technologies and standards can be found in [4] and [1]. Cyber security and privacy issues in Smart Grid are discussed in [10] and [13]. Stealthy attacks on SCADA systems in Power Networks are studied in [14] and [2].

Distributed consensus protocols are an important class of distributed algorithms. The goal is to find an agreement between all nodes without having any centralized unit. In distributed consensus protocols, consensus is found based on the values exchanged between immediate neighbors. Consensus problems and conditions for reachability of consensus have been intensively studied in [12] and [11].

Resilient Distributed Consensus is a special case of distributed consensus. It is assumed that a limited number of malicious nodes are present that try to disrupt consensus by providing false status information to their neighbors. In [7, 9] a fully distributed Adversary Resilient Consensus Protocol (ARC-P) has been introduced. The topological properties necessary for reaching consensus with ARC-P have been studied in [8]. As we extend this protocol, we present it in more details.

ARC-P operates on mesh networks. The basic assumption is that the number of malicious nodes is bounded: there are at most $FTotal$ malicious nodes in the whole network and at most $FLocal$ malicious nodes in the neighborhood of every genuine (i.e., non-malicious) node. Based on this assumption, the ARC-P algorithm proceeds as follows. First, it sorts the status values from all neighboring nodes. Then it removes up to FLocal largest as well as up to FLocal smallest values, which are strictly smaller or respectively larger than the node's own value. After that, it computes the average of the remaining values (including its own value) and assigns this as the new value of the node.

Please note that ARC-P makes no assumptions about trustworthiness of any neighboring nodes. Due to the necessity for multiple information flow paths, ARC-P can only reach consensus in a mesh network topology.

3. PROBLEM FORMULATION

In this section, we first describe the Building Area Network (BAN) scenario and then present the considered threat model. Throughout this section we present arguments justifying our assumptions.

3.1 Scenario: Building Area Network (BAN) in Smart Grid

The Home Area Network (HAN) and Building Area Network (BAN) are important concepts in Smart Grid. They provide an envelope for the interconnection between Smart Appliances (SA) (e.g., washer, air conditioner, etc.) and Smart Metering Infrastructure (SMI). Theoretically, it is possible to provide redundant network connections between SA and SMI. However, economic considerations, such as costs of infrastructure, development, installation, and maintenance, are likely to prevent such solutions in HAN and BAN areas. Consequently, the physical network topology will be sparse, but sufficient for the purpose network infrastructure, and will resemble a tree topology (see Figure 1).

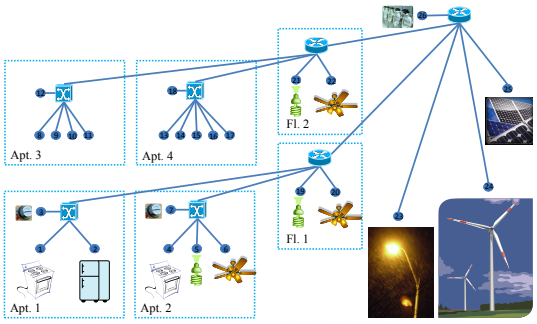

Figure 1: Building Area Network (BAN) Network Topology

Assuming the state of the art "plug and play" scenarios with auto-discovery capabilities, the logical topology of interconnection between nodes will resemble the physical topology with all communicating nodes arranged in a tree (see Figure 2). The numbers within nodes in both figures are voluntarily assigned IDs of all SA/AMI nodes. Numbers near nodes in Figure 2 are arbitrarily selected initial values, which will be discussed in more detail in Section 5.

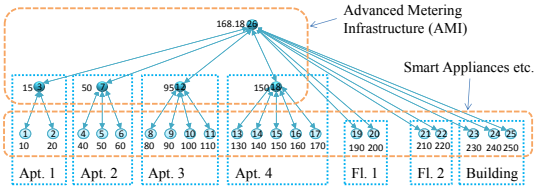

Figure 2: Building Area Network (BAN) Overlay Topology

Coordination between different SA is seen as one of the advantages of the Smart Grid, which, for instance, can reduce energy spikes caused by simultaneously turning appliances on or off. Such coordination between SA can be realized in a distributed manner, in which various SA exchange information among themselves either directly or indirectly via SMI. We assume that there will be no direct communication between SA for the following reasons. Technically, coordination between different classes of SA eventually produced by different manufacturers is a much harder problem than coordination via AMI with the standardized interface. Furthermore, within a single household such a solution remains very scalable as the number of SA is limited. Last, but not least, communication between SA from different households is not likely to be allowed because of privacy and security concerns.

3.2 Threat Model

In an environment like Smart Grid, or its constituent part BAN, there is a clear distinction between the exposure of its various components to attacks. For instance, physical access to AMI is usually restricted to authorized personnel. Usually, there are also mechanisms in place to detect physical or cyber manipulations on AMI.

On the other hand, various SA are owned by the residents who also have unrestricted physical access to it. This makes SA exposed to various manipulations. Moreover, as the number or the type of SA is neither restricted nor controlled by the electricity provider, plugging in manipulated devices as well as computers impersonating SA can be seen as a plausible and cost-effective attack vector.

Consequently, we consider the following threat model. Only leaf nodes (which represent various SA) in the connectivity tree can be malicious. All non-leaf nodes (which represent AMI) are considered to be genuine and behave according to the consensus algorithm. The only attack we consider in this work is injection of the false information by the malicious nodes.

We are well aware that it is possible to perform at least two further classes of attacks. First, infection of AMI, e.g., via exploiting a buffer overflow attack. A second attack is DoS or DDoS attack either on the AMI or on SA. We explicitly omit consideration of these two classes of attacks for the following reasons. A buffer overflow attack requires significantly greater effort than injection of false information by compromised SA. Furthermore, AMI can be hardened to detect and to collect evidence of an attempted buffer overflow attack, thus opening the possibility of legal prosecution. Unlike the code injection attacks, DoS and DDoS attacks can be detected easily.

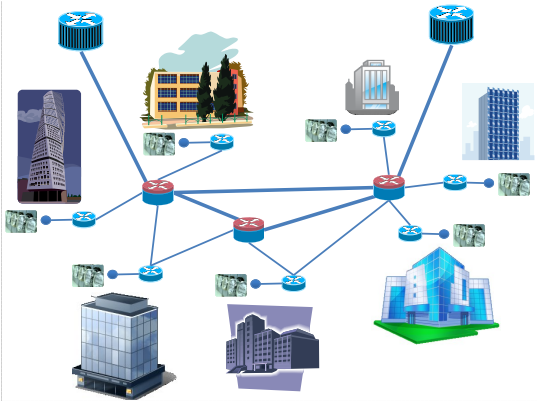

Figure 3: Neighborhood Area Network (NAN) Network Topology

In the present work, we also omit consideration of attacks on network components, e.g., on routers. Even though such attacks are possible, their complexity as well as resources required are way beyond those required for attacks on SA. This is especially true for the network infrastructure to which BAN is connected, e.g., the Neighborhood Area Network (NAN). At this level, it is common to have a core network with redundant physical topology (a schematic example is depicted in Figure 3), which is robust against both infrastructure failure and attacks on it.

3.3 Terms and Definitions

For the definition of various forms of consensus we use in this paper, we have to introduce the following terms for the tree $G = \{V, E\}$:

$V = V^{SA} \cup V^{SMI} : V^{SA} \cap V^{SMI} = \emptyset$

$V^{SA} = V^{g} \cup V^{m} : V^{g} \cap V^{m} = \emptyset$

Here, V^{SA} are all leaf nodes representing SA, V^{SMI} all non-leaf nodes representing SMI, V^{g} all genuine leaf nodes, and V^{m} all malicious leaf nodes. We also introduce the

function $\pi(v_i)$, which returns the parent node of any non-root node v_i and which returns v_i if it is the root.

For this work, we are only interested in siblings among leaf nodes. We define leaf siblings as

$\widetilde{V_k} = \{v_i : \forall v_i \in V^{SA} \wedge \pi\{v_i\} = v_k\}$

In the considered scenario, Smart Grid leaf nodes, i.e., all kinds of SA, are the only active nodes. All non-leaf nodes, which represent Smart Metering Infrastructure, act rather as mitigating and filtering nodes.

We distinguish between following cases of consensus:

Global Consensus: We define *Global Consensus* as a consensus among all genuine leaf nodes. In mathematical terms,

$\forall v_k(0)\ \exists t : v_i^g(t) = v_j^g(t)\ \forall v_i^g, v_j^g \in V^g$,

where $v_k(0)$ represents initial value of any node, including non-leaf and malicious ones, and $v_i^g(t)$ and $v_j^g(t)$ are values of any two genuine leaf nodes after t rounds of a consensus algorithm. Please note that this definition emphasizes that there are no restrictions imposed on the initial values of any node, including non-leaf ones.

Weak Partial Consensus: We define *Weak Partial Consensus* as a consensus between all genuine nodes in at least one sibling neighborhood, i.e., sharing the same parent node:

$\exists \widetilde{V_k}, t : \forall v_i, v_j \in \widetilde{V_k} \cap V^g : v_i^g(t) = v_j^g(t)$

Strong Partial Consensus: We define *Strong Partial Consensus* as a consensus between all genuine nodes within at least two leaf neighborhoods:

$\exists \widetilde{V_k}, \widetilde{V_q}, t : \forall v_i, v_j \in (\widetilde{V_k} \cup \widetilde{V_k}) \cap V^g : v_i^g(t) = v_j^g(t)$

No Consensus: We say that *No Consensus* can be reached if no Weak Partial Consensus can be reached. We also speak about absence of consensus in the case where malicious node(s) can determine the consensus values of all genuine nodes.

We further distinguish between $FTotal$ and $FLocal$ values. Whereas FTotal is the overall number of malicious leaf nodes in the tree, FLocal is a reasonable assumption on how many nodes in the neighborhood can provide malicious information (because they are malicious or because they were outvoted by malicious nodes). FLocal acts as a parameter for the original ARC-P algorithm as well as for the R.ARC-P algorithm we will present in Section 4. We further assume that FLocal assumption can be wrong. Therefore, by the parents of leaf nodes, the maximal number of malicious nodes can reach in extreme case FTotal value.

4. RANGE-BASED ARC-P

Based on our assumption that only leaf nodes can be malicious, we can assume with the high confidence that parent nodes generally provide (more) genuine information and only child nodes can provide incorrect information. This allows us to extend the ARC-P by always trusting the value of a parent node, i.e., never exclude this value alongside with a node's own value. In our proposal, Range-based ARC-P (R.ARC-P), we go one step further. For every node, we declare that the range of values between the node's own value and its parent's value is valid. Selection of the valid values in R.ARC-P is graphically depicted in Figure 4.

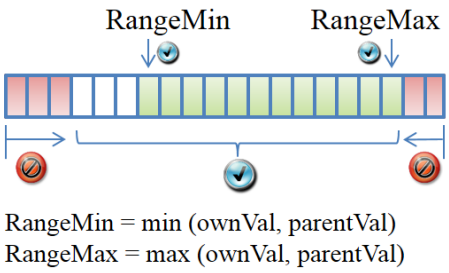

Figure 4: Selection of valid values in R.ARC-P

A C-like pseudo code of R.ARC-P executed by every node in the tree is depicted in Figure 5. The R.ARC-P algorithm works as follows. First, the minimum and maximum values of the – for this particular node in this particular round – always valid value range ($RangeMin$ and $RangeMax$) are defined as the minimum and maximum of its own and parent values, respectively. After sorting values from all peers (including its own and parent values), up to $FLocal$ smallest nodes are removed, as long as they are strictly smaller than $RangeMin$, and up to $FLocal$ largest values are removed, as long as they are strictly larger than $RangeMax$. In the pseudo code, we use indices to the smallest ($iRangeMin$) and biggest ($iRangeMax$) valid value as the means of "removal." After that, an average of all valid values is computed. Please note that $arrPeerVals$ should contain all values from all neighbors (including parent node) and its own value.

```
RARCP (arrPeerVals, ownVal, parentVal, FLocal)
{
  RangeMin = min (ownVal, parentVal);
  RangeMax = max (ownVal, parentVal);

  sort (arrPeerVals);

  iRangeMin = 0;
  for (i=0; i<FLocal; iRangeMin++, i++)
    if ((arrPeerVals[iRangeMin] >= RangeMin)
      break;

  iRangeMax = sizeof(arrPeerVals)-1;
  for (i=0; i<FLocal; iRangeMax--, i++)
    if ((arrPeerVals[iRangeMax] <= RangeMax)
      break;

  newVal = 0;
  for (i=iRangeMin; i<iRangeMax; i++)
    newVal += arrPeerVals[i];

  newVal /= (iRangeMax-iRangeMin+1);

  return newVal;
}
```

Figure 5: R.ARC-P Algorithm

5. EVALUATION

In this section, we first describe the settings for the experiments and later present some of test results.

Malicious Nodes		Reachability of Consensus		
FLocal	FTotal	ARC-P	ARC-P^d	R.ARC-P
0	0	√	√	√
0	1	–	–	–
1	0	–	(–)	√
1	1	–	(–)	√
1	2	–	(–)	(√)
2	0	–	(√)	√
2	1	–	(√)	√
2	2	–	(√)	√
2	3	–	(–)	(√)
2	5	–	(–)	(√)

√ Global Consensus
(√) Strong Partial Consensus
(–) Weak Partial Consensus
– No Consensus

Table 1: Evaluation summary

5.1 Evaluation Setup

We have compared capabilities of ARC-P and R.ARC-P to reach consensus on different tree topologies with different initial values of all nodes. Furthermore, in order to overcome the obvious restriction of ARC-P in the case of leaf nodes (i.e., if FLocal is set to a value greater than zero no information will flow towards the leaf), we have evaluated how ARC-P with different FLocal values for leaf and non-leaf nodes will perform. We call this variant ARCP^d ("d" for **d**ifferentiated settings). In ARCP^d all non-leaf nodes perform like original ARC-P with the specified FLocal, for leafs FLocal is always set to 0.

For the sake of simplicity, all experiments presented in this paper are run on the topology depicted in Figure 2. In this figure, numbers within nodes indicate the arbitrarily selected IDs of the nodes. Before the consensus algorithm starts, we have initialized all leaf nodes with the value $10 * NodeID$. All non-leaf nodes are initialized with average values of their children nodes. In Figure 2, all initial values are depicted near nodes. Simulation experiments have been performed with MATLAB.

In our experiments, we have varied the total number of malicious nodes from zero to five, in all possible combinations of their placement in leaf nodes. Independently of the number of nodes, we have also varied the number of expected malicious nodes from zero to three. During simulation, all genuine nodes update their own value according to the selected protocol, all malicious values do not deviate from the initial value over time. This behavior simplifies visual distinction on the graphics between values of genuine and malicious nodes (values of malicious nodes are depicted as straight horizontal lines). Not presented in this section, we have also performed experiments with the values of malicious nodes changing periodically. Results regarding reaching consensus of different algorithms and their variations are not influenced by these fluctuations.

Please note that in this work we assume that all neighboring nodes exchange actual values, i.e., we don't consider the case when malicious nodes can provide different status values to different neighbors.

5.2 Simulation Results

The summary of various experiments is summarized in Table 1. The leftmost two columns specify FLocal and FTotal numbers of adversaries during the experiment, i.e., the expected (by the algorithm) maximal number of malicious nodes in the neighborhood and the real number of malicious nodes in the tree. The results of experiments are depicted in the remaining three columns for ARC-P, ARC-P^d, and R.ARC-P protocols respectively. The results reflect the worst case scenario, i.e., location of malicious nodes under which the particular protocol performs the worst. Please note that we assume that FLocal expectation can be wrong. Therefore, the worst case scenario can reflect the situation when up to FTotal malicious nodes are in the neighborhood of a single node. We distinguish between the following four cases of protocol performance: Global Consensus, Strong Partial Consensus, Weak Partial Consensus, and No Consensus (see Section 3.3 for the definition of these terms).

In the case when there are no malicious nodes in the system and none are expected, all ARC-P, ARC-P^d, and R.ARC-P perform similarly and produce identical convergence behavior (see Figure 6).

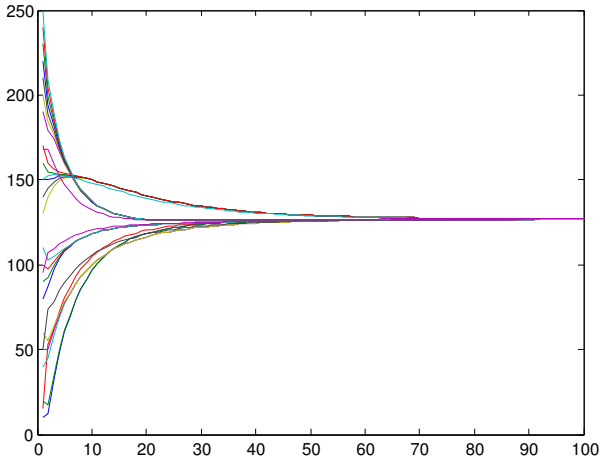

Figure 6: ARC-P, ARC-P^d, and R.ARC-P with FLocal=0, FTotal=0

In the case when there is one malicious node and none is expected, ARC-P, ARC-P^d, and R.ARC-P algorithms converge as well, but this time to the value of the malicious node (see Figure 7). Please note that in this figure, unlike all other presented in this paper, 500 rounds are depicted.

In the case when nodes expect that one of their neighbors is malicious, ARC-P cannot reach consensus even if there are no malicious nodes in the tree (see Figure 8). The reason is simple. As leaf nodes are only connected to their parents, there is no other information source which can be removed from the considerations nor are there any further information flow paths which would "feed" information to the node.

The ARC-P^d configuration of ARC-P performs slightly better (see Figure 9) - it is capable to reach Weak Partial Consensus for all sub-trees representing apartments of BAN as well as for the remaining SA of the building.

Under the same conditions, the proposed R.ARC-P algorithm is able to reach consensus among all nodes in the tree

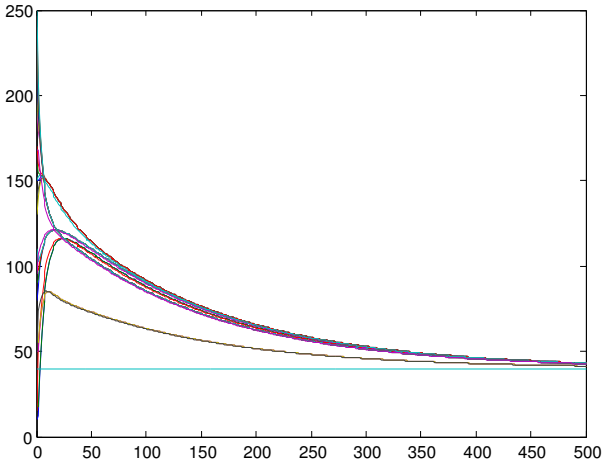

Figure 7: ARC-P, ARC-P^d, and R.ARC-P with FLocal=0, FTotal=1 (node 4)

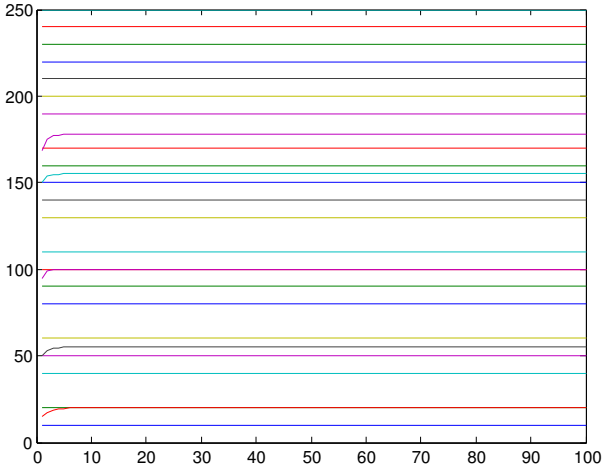

Figure 8: ARC-P with FLocal=1, FTotal=0

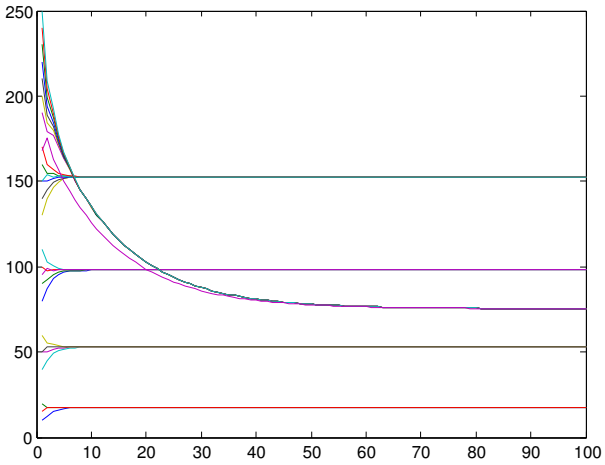

Figure 9: ARC-P^d with FLocal=1, FTotal=0

(see Figure 10). The reason is that all leaf nodes are trusting parents and – if they are genuine - their own value will always asymptotically converge to the value of their parent nodes.

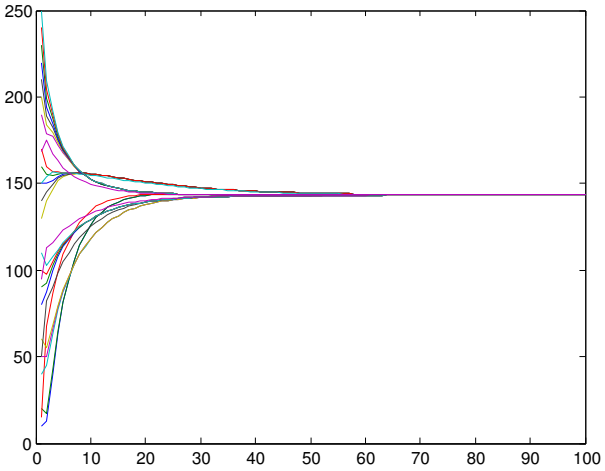

Figure 10: R.ARC-P with FLocal=1, FTotal=0

As even expectation of a single malicious node in the neighborhood prevents ARC-P protocol from converging, we will not present further examples with higher real and higher FLocal and FTotal.

Throughout our experiments, we have seen that R.ARC-P converges in all cases if the number of expected malicious nodes is correct, i.e., identical or greater than the real number of malicious nodes in the neighborhood. For instance, in Figure 11 an example is present with up to 3 malicious nodes in neighborhood, even though totally 5 malicious nodes are present in the system. R.ARC-P shows very fast convergence of all genuine nodes to the same value.

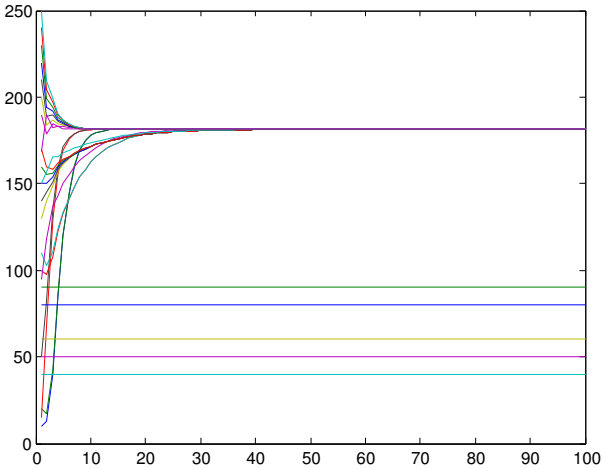

Figure 11: R.ARC-P with FLocal=3, FTotal=5 (malicious nodes: 4, 5, 6, 8, and 9)

Please note that, even under the same graph topology and initial values of nodes, the ultimate convergence value can vary influenced by the number of expected malicious nodes. This convergence error is introduced by removing outlier

values from the computation of a new state value. The result is identical with the behavior of ARC-P protocol and is a "price" paid for the resilience against false information injection.

6. DISCUSSION

In this section, we first provide a brief discussion about underlying graph theoretical reasons for the different performance of the evaluated distributed consensus protocols. Then we outline ideas how the presence of false information can be detected and how malicious nodes can be localized if R.ARC-P protocol is used.

6.1 Reasoning about R.ARC-P Performance

Considering a generic graph in which ARC-P, ARC-P^d, and R.ARC-P tries to find a consensus can be seen as an information flow problem in a dynamic digraph, i.e., directed graph (for the related discussion see, e.g., [6, 5]). Exclusion of values of some neighbors is equivalent to the removing in edges comming from these nodes. Therefore, if ARC-P is applied in the tree with $FLocal \geq 1$, no information can flow towards leafs because edges from their parents are removed.

In the ARC-P^d, setting $FLocal = 0$ for all leafs fixes this problem only to some extent. This ensures that information is always flowing from the leaf parents to their leafs, thus creating preconditions for the Weak Partial Consensus around values of the leaf parents. However, indiscriminate treatment of all neighbors by non-leaf nodes enables removal of edges which makes graph disconnected, making global consensus impossible.

R.ARC-P fixes this problem by introducing hierarchical trust relationships between nodes. This alone ensures that there always exists a spanning tree in a digraph, starting at the root of the tree in which consensus should be reached. This, in turn, ensures that even in the worst case scenario all genuine nodes converge to the value of the root node. The introduction of the valid range (between own and parent node values) decreases the number of edges to be removed, thus fostering the "upstream" information flow from leafs to the root. Furthermore, the presence of the values in the range between own and parent node values, their number, and their values influence the convergence speed of a node's own value towards the value of the parent node. Therefore, R.ARC-P always reaches Global Consensus if FLocal is estimated correctly and tends to reach Strong Partial Consensus if the FLocal estimation was wrong.

6.2 Localization of Malicious Nodes

As we have thought during simulations, the R.ARC-P will not converge to a single consensus value if the number of malicious nodes in the neighborhood is underestimated. However, we would like to emphasize an interesting convergence behavior in this case. Figure 12 depicts the state development with nodes 5 and 6 malicious (i.e., FTotal=2) and expected number of malicious nodes FLocal=1.

According to the experimental setup, malicious nodes keep their own values unchanged. The genuine node 4 will converge to the value of its parent node 7. The node 7, however, will "stabilize" at the value between node 6 (an malicious nodes which considered genuine because of the wrong estimation) and node 26 (a parent node). This observation allows us to perform detection and – to some extent – localization of the malicious node(s). We distinguish between the following cases:

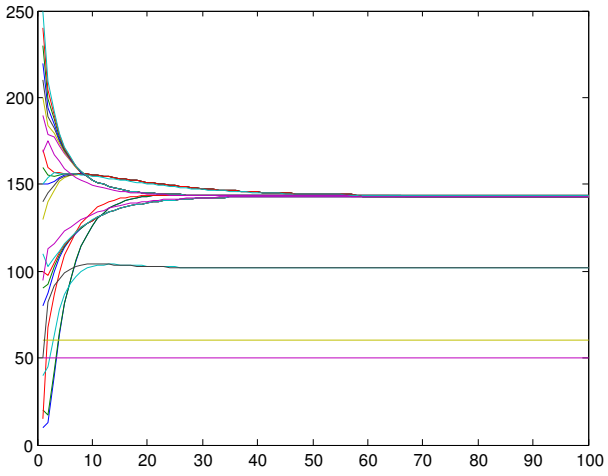

Figure 12: R.ARC-P with 1 malicious nodes expected and 2 present (nodes 5 and 6)

- A parent node can monitor the behavior of the children leaf nodes. If they refuse to converge towards the value of the parent node, these nodes can be identified as malicious.
- In the case if child node, which is not a leaf node, does not converge, the assumption can be made that this child node was outvoted by an exceeding number of malicious children nodes.
- In the case of self-observation of a non-leaf node, its inability to converge with the parent node indicates that the number of malicious children is higher than expected.

These observations have two consequences. First of all, it is possible to identify misbehavior and – to some extent – localization of malicious node(s). Second, it opens the possibility for a dynamic adaptation of the number of expected malicious nodes in the neighborhood. We plan to evaluate both of these possibilities in more details in our future work.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a novel resilient distributed consensus protocol, Range-based Adversary Resilient Consensus Protocol (R.ARC-P). Compared to the ARC-P protocol it extends, R.ARC-P has three main differences from ARC-P. First, whereas ARC-P requires mesh topology in order to reach consensus, R.ARC-P operates on the tree topology. Second, ARC-P is an indiscriminate protocol, treating all nodes in the neighborhood equally; R.ARC-P is a discriminating protocol introducing different trust relationships based on the child-parent relation in the tree topology. Third, in ARC-P, the only value that is always trusted is a node's own value; in R.ARC-P, the range of values between a node's own value and the value of the parent node is considered as trustworthy. With the simulation results we have shown that R.ARC-P can always reach Global Consensus if FLocal assumption is correct and can reach Strong Partial

Consensus if it is not. We have discussed the graph theoretical reasons behind this behavior. Furthermore, we have outlined a procedure allowing localization of malicious nodes if R.ARC-P algorithm is used.

In our future work, we plan to investigate the consensus problem in scale-free networks, which can be seen as a combination of mesh and tree sub-networks. An immediate assumption is that the original ARC-P should be applied for in the mesh part of such networks whereas the R.ARC-P protocol proposed in this paper in the tree sub-networks. We plan to evaluate the interplay of these two protocol variations.

Last but not least, even though in this work we have omitted discussing infection and DoS attacks, we are planning to investigate the influence of such attacks on the distributed consensus algorithm. Most interestingly, we are planning to analyze the question whether a combination of application and network level attacks can give an adversary a significant advantage, and, if it is the case, under what conditions.

Acknowledgments

This work is supported in part by the National Science Foundation (CNS-1238959, CNS-1035655) and NIST (70NANB13H169).

8. REFERENCES

[1] F. Cleveland. Iec tc57 security standards for the power system's information infrastructure-beyond simple encryption. In *Transmission and Distribution Conference and Exhibition, 2005/2006 IEEE PES*, pages 1079–1087. IEEE, 2006.

[2] G. Dán and H. Sandberg. Stealth attacks and protection schemes for state estimators in power systems. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 214–219. IEEE, 2010.

[3] J. Gao, Y. Xiao, J. Liu, W. Liang, and C. Chen. A survey of communication/networking in smart grids. *Future Generation Computer Systems*, 28(2):391–404, 2012.

[4] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke. Smart grid technologies: communication technologies and standards. *Industrial informatics, IEEE transactions on*, 7(4):529–539, 2011.

[5] P. Holme. Network reachability of real-world contact sequences. *Physical Review E*, 71(4):046119, 2005.

[6] P. Holme and J. Saramäki. Temporal networks. *Physics reports*, 519(3):97–125, 2012.

[7] H. J. LeBlanc and X. D. Koutsoukos. Consensus in networked multi-agent systems with adversaries. In *Proceedings of the 14th international conference on Hybrid systems: computation and control*, pages 281–290. ACM, 2011.

[8] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram. Resilient asymptotic consensus in robust networks. *Selected Areas in Communications, IEEE Journal on*, 31(4):766–781, 2013.

[9] H. J. LeBlanc, H. Zhang, S. Sundaram, and X. Koutsoukos. Consensus of multi-agent networks in the presence of adversaries using only local information. In *Proceedings of the 1st international conference on High Confidence Networked Systems*, pages 1–10. ACM, 2012.

[10] J. Liu, Y. Xiao, S. Li, W. Liang, and C. Chen. Cyber security and privacy issues in smart grids. 2012.

[11] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[12] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *Automatic Control, IEEE Transactions on*, 49(9):1520–1533, 2004.

[13] E. Quinn. Privacy and the new energy infrastructure. *Available at SSRN 1370731*, 2009.

[14] H. Sandberg, A. Teixeira, and K. H. Johansson. On security indices for state estimators in power networks. In *Preprints of the First Workshop on Secure Control Systems, CPSWEEK 2010, Stockholm, Sweden*, 2010.

[15] N. Saputro, K. Akkaya, and S. Uludag. A survey of routing protocols for smart grid communications. *Computer Networks*, 56(11):2742–2771, 2012.

[16] C. Soto, B. Song, and A. K. Roy-Chowdhury. Distributed multi-target tracking in a self-configuring camera network. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1486–1493. IEEE, 2009.