European Research Infrastructure supporting Smart Grid Systems Technology Development, Validation and Roll Out

Technical Report TA User Project

OpenDISCO - Open Source Security Assessment Framework for Distributed Control in the Smart Energy Grid

Grant Agreement No:	**654113**
Funding Instrument:	**Research and Innovation Actions (RIA) – Integrating Activity (IA)**
Funded under:	**INFRAIA-1-2014/2015: Integrating and opening existing national and regional research infrastructures of European interest**
Starting date of project:	**01.11.2015**
Project Duration:	**54 month**

Contractual delivery date:	**03.09.2018**
Actual delivery date:	**03.09.2018**
Name of lead beneficiary for this deliverable:	**Prof. Dr. Hannes Federrath**
Deliverable Type:	**Report (R)**
Security Class:	**Public (PU)**
Revision / Status:	**draft**

Project co-funded by the European Commission within the H2020 Programme (2014-2020)

Document Information

Document Version: 1

Revision / Status: draft

All Authors/Partners Universität Hamburg, University of Strathclyde

Distribution List

Document History

Revision	Content / Changes	Resp. Partner	Date
2	Document formatting, table of content update and minor edit of sections 2, 3, 4	USTRATH	10.01.19

Document Approval

Final Approval	Name	Resp. Partner	Date
[Review Task Level]	[Given Name + Name]	[Partner Short Name]	DD.MM.YY
[Review WP Level]	[Given Name + Name]	[Partner Short Name]	DD.MM.YY
[Review Steering Com. Level]	[Given Name + Name]	[Partner Short Name]	DD.MM.YY

Disclaimer

Neither the Trans-national Access User Group as a whole, nor any single person warrant that the information contained in this document is capable of use, nor that the use of such information is free from risk. Neither the Trans-national Access User Group as a whole, nor any single person accepts any liability for loss or damage suffered by any person using the information.

This document does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of its content.

Copyright Notice

Table of contents

Abbreviations

DER	Distributed Energy Resource
TA	Trans-national Access
ISO	*International Standardization Organisation*
OSI model	Open Systems Interconnection model, describing the layers of network communication (physical layer, data link, network layer, transport layer, session layer, presentation layer and application layer)
DCA	Distributed Control Algorithm
DER	Distributed Energy Resource

Executive Summary

The Security and Privacy working group at the Department for Computer Science at the University of Hamburg is currently developing a security assessment framework for distributed Smart Grid control. To elaborate on the capabilities of said framework with the working title “OpenDISCO”, Transnational Access (TA) at the Smart Grid Lab of the University of Strathclyde in Glasgow has been arranged. During the TA, a custom implementation of a pre-existent distributed Smart Grid control algorithm has been analysed in terms of resilience against cyber-attacks (Denial-of-Service attacks). These attacks at the application level of the ISO OSI architecture include a) dropping of messages, b) delay of messages, c) deactivation of nodes and d) deactivation of communication lines.
A systematic approach has been chosen to evaluate the effect of application layer delays of an extent between 0ms and, in steps of 500ms, 10 seconds. The evaluation shows that the framework can ease the task of evaluating the resilience of distributed Smart Grid control algorithms. This has been exemplarily shown in the case of denial-of-service attacks. The investigated control algorithm has been shown to be robust towards overshoots for delays up to 3 seconds, and for greater delays to be partially unable to recover the nominal frequency of 50 Hz.
The results will be used to elaborate general measures for enhancing distributed control algorithms. In future work, these general measures will be evaluated in a comparable setup.

1 General Information of the User Project

Title: OpenDISCO - Open Source Security Assessment Framework for Distributed Control in the Smart Energy Grid
Host Infrastructure: Dynamic Power Systems Lab of the University of Strathclyde in Glasgow
Access period: 12 days (8 access days) between 3. Sept and 5. Oct 2018.
The User Group consists of:
Marius Stübs (Doctoral Researcher / Local Group Leader)
Kevin Köster (Assistant Researcher)
Prof. Dr. Hannes Federrath (Supervisor, Full Professor at Universität Hamburg)
The University of Strathclyde provided active support for the project delivered by Paulius Dambrauskas and Mazheruddin Syed.

2 Research Motivation

The project group from the University of Hamburg is currently developing a distributed security assessment framework with the focus on Smart Grid applications.
The main objective of the transnational laboratory access was to show the capabilities of the framework in a realistic scenario. The experiment was set up as a controller-hardware-in-the-loop environment, thereby investigating the effects of denial-of-service attacks on a reference implementation of distributed scheduling scheme, controlling the frequency of an islanded microgrid simulated on a real-time digital simulator, as configured with RSCAD.
Regarding the methodology, the distributed framework nodes were configured to apply denial-of-service attacks of increasing severity (1ms to 10 seconds). The effect of the cyber-physical system was recorded. One distinguishing feature of the framework is its decentralization: Thus, the investigated effects where introduced independently on every node in the network, enabling high scalability of the approach.

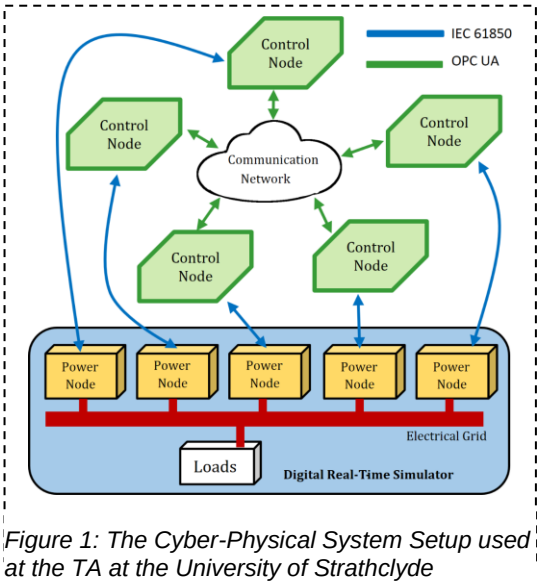


Figure 1: The Cyber-Physical System Setup used at the TA at the University of Strathclyde

2.1 Objectives

Structure of a Single Control Node
The proposed framework provides an easy-to-use API for integrating different functionalities as software modules into the framework's core.
The data flow within a node is realized using a message queue. Any subscribed value from a different OPC UA server is inserted into the message queue and any message given to the queue can be accessed by the present node modules using filters, introducing an elegant and fast way of reacting to events. The nodes implement a client and a server module. In each iteration, the node's server module writes the computed temporary consensus suggestion in the OPC UA Database. Each client subscribes to all their neighbors' server's OPC UA Database. If that value changes, the OPC UA server informs all subscribers of the change.
The communication stack of OPC UA assumes a separation of client and server applications. The server side is traditionally only responding to commands, like a sensor or a motor. The client is usually located in a control room, assessing status data from the servers, execute a centralized algorithm and sending commands to the servers. To decentralize an algorithm, each node in the network is extended by a

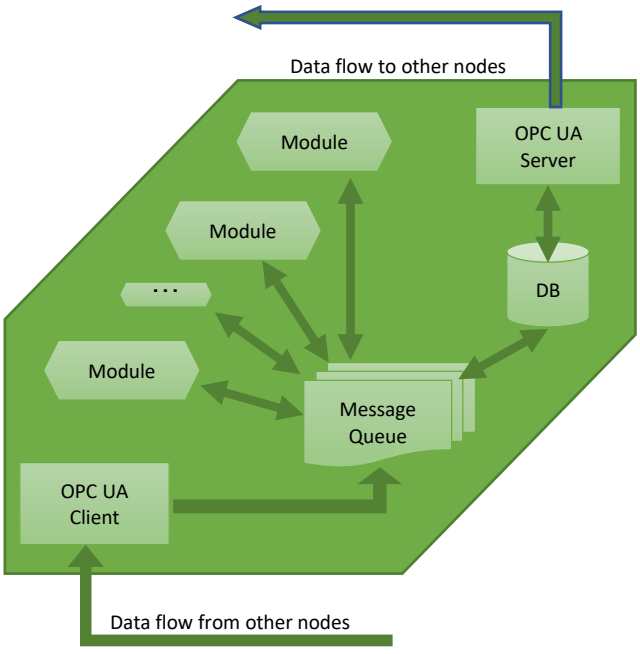


Figure 2: Internal structure of a single Control Node. New messages from other Control Nodes are collected by the OPC UA Client and distributed via an internal Message Queue. Computed values are stored in a database and then distributed via an OPC UA server to other Control Nodes.

module representing a client. Thereby, each node can provide information about its status via their server module, while accessing the other nodes using its client module.

Experiment 1 – Distributed Control

For our first experiment, we regard the perspective of distributed frequency control. The goal of the control structures is to maintain frequency stability of the nominal 50 Hz grid frequency. This is achieved by two mechanisms. First, we simulated a distributed droop-based P-control algorithm running at the Energy Devices. Thereby, in case of a disturbance, the frequency can be locally stabilized in each Energy Device, but the overall system frequency deviated from the nominal frequency. In the next step, the nominal frequency has to be gradually restored. This requires the Control Nodes to communication and find a consensus. As soon as the consensus is reached, each Control Nodes sends the appropriate commands to their respective Energy Device.

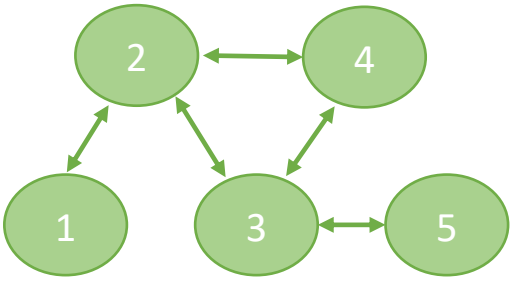


Figure 3: Exemplary topology of the peer-to-peer communication links between the control nodes.

As a first application of the distributed control by the control nodes, we have implemented a distributed consensus algorithm [1] using our framework. We have simulated this setup in software and found positive indication that frequency recovery can be achieved within seconds. As shown in Figure 5, the consensus algorithm on the five control nodes generally converges within about a second and a half, with a total of needed six iterations to recover 50 Hz and a total runtime of about 15 seconds.

```
<Attack FormatVersion="1">
    <Effects>
        <Send>
            <Delay>
                <Periodicity>1</Periodici
                <Probability>0.2</Probabi
                <ms>200</ms>
            </Delay>
        </Send>
    </Effects>
</Attack>
```

Figure 4: Exemplary stress condition configuration via XML file. For each transmitted message, it is probabilistically delayed for 200 ms with a chance of 20%.

To verify these results using a real-time digital simulator and control-hardware-in-the-loop was the goal of our first experiment at the smart grid lab at Strathclyde University.

Experiment 2 – Availability under Stress

The second experiment aims on the robustness of the proposed setup. Since distributed consensus is dependent of message distribution, we are investigating the impact of a network "under stress". To accomplish such conditions, we have prepared a specific module to our framework, that deliberately loses or delays messages. Thereby we can simulate denial of service attacks as well as malfunctioning nodes. Our interest lies mainly in the behavior of the OPC UA implementation used, open62541 [2,3], and experimental indication on how to systematically improve its performance. As seen in Figure 4, each single stress condition ("attack") can be described independently from the actual implementation in an XML style file. Figure 4 shows the configuration of a delay effect, where each sent message is possible delayed by 200ms, where the effect has a probability of 20% to be carried out.

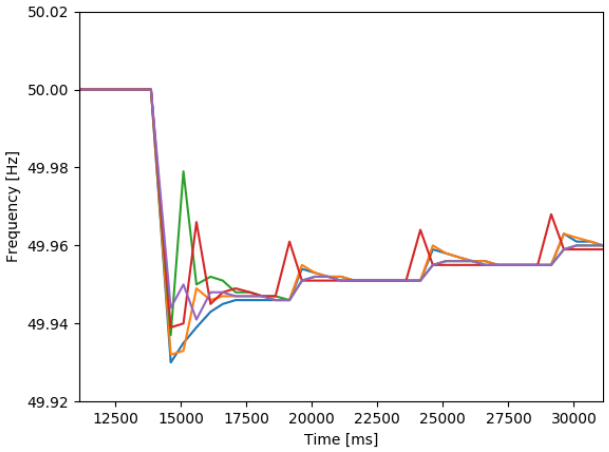


Figure 5: An example run of the proposed framework with the selected DCA for load-frequency stability with an artificial delay introduced. After the disturbance, in second 14 the nominal frequency of 50 Hz is being restored more slowly than expected. Each line with a different colour represents the consensus computation of a single DER / control node.

As seen in Figure 5, forcing a delay for single connections between control nodes leads to a significant delay in the overall "consensus finding

time".
From the experiments with a real-time digital simulator we expect to find different response patterns. This would allow us to adjust our strategies to improve the robustness of the used DCA and might even lead to a generalizable description of response patterns.

2.2 Scope

Our setup utilizes the traditional load-frequency-control scenario where the frequency of the power system needs to be stabilized and then smoothly recovered to the nominal 50 Hz value. The setup consists of several different parts. On the top level, there are **Control Nodes**, realized by five ARM-based embedded devices running our proposed framework. This part is depicted by the orange boxes in Figure 3. Their interconnection is part of the physical setup and we use a standardized network router to implement it, represented also in Figure 3 by the cloud symbol.
Likewise to the control nodes, on the electrical grid we simulate five **Power Nodes**, for example battery storages, depicted as yellow boxes in figure 3, each connected via a power inverters to the **AC grid**, represented by the red line. Also connected to the **AC Grid** we simulate an **Electrical Load** (white box), that can be switched on or off.
Each Control Node is exclusively responsible for the control of an Energy Device using a direct link, shown as a blue line in Figure 3.

Primary Control vs. Secondary Control

In our setup, we differentiate between primary control and secondary control. The usage of these terms in this proposal are inspired by, but not equal, to the wording in the Operating Reserve of electricity networks.

Primary control is a function that is run on each Power Node and does not require communication at all. By decreasing (increasing) the output frequency at higher (lower) active power consumption, it regulates the load distribution between the Power Nodes, while the resulting grid frequency might deviate from its nominal value. Figure 4 shows an exemplary frequency drop following in a load step.

Secondary Control on the other hand is located in the Control Node. Its duty is to smoothly recover the nominal frequency of 50 Hz, subsequently after the frequency is stable on a non-optimal level. Since all Control Nodes act simultaneously, it requires communication and is realized utilizing a consensus algorithm. As soon as all control nodes agreed upon their action, each Control Node sends the appropriate command to its respective Energy Node.

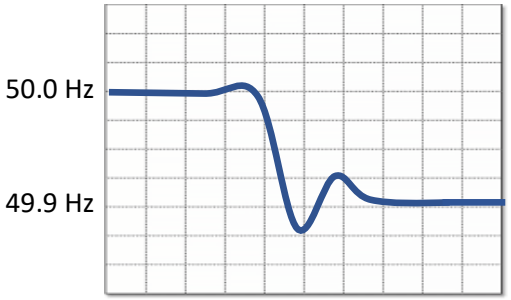


Figure 6: The term Primary Control in this scenario describes the process of stabilizing the frequency of the power system at a possibly non-optimal level. In this figure, first the frequency drops due to an event on power system level and is then stabilized on 49.9 Hz within milliseconds.

Structure of a Single Power Node

The Power Nodes are part of the physical setup. Each Power Node consists of a battery storage, a power inverter and a logical unit for the primary control. The primary control utilizes the droop control strategy and provides the reference points for the voltage and current control loops to adjust power output and stabilize system using (1),

$$f = f_0 - k_P (P - P_0)$$
$$V = V_0 - k_Q (Q - Q_0) \quad (1)$$

where the variables are defined by (2) as:

P_0; Q_0:	the normal value of active and reactive power
P; Q:	the active and reactive power supplied by DG
f_0:	the rated frequency
V_0:	the rated amplitude of grid voltage
f:	the actual measured values of frequency

(2)

V: the actual measured values of voltage
k_P: the active power coefficient
k_Q: the reactive power coefficient

All of inverter based DGs working as grid-forming type contribute to maintain the voltage and frequency stability and keep it close to nominal values. The frequency and amplitude deviations will be eliminated in secondary control level. The droop control strategy can maintain power output and frequency among the participants of the microgrid in an immediate, communication-less way. Since each inverter has its own droop coefficient k_P and k_Q and rated power, the resulting power levels will differ depending on the deviation.

3 State-of-the-Art

When deploying new control structures, protocols and algorithms, simulation is a reasonable midway for evaluating their robustness whilst minimizing the necessary effort [4] and balancing it regarding realistic setup and generalizability of the findings. There already exists a variety of tools for simulating attacks on distributed networks. This includes tools that evaluate the security of power systems. The need for power system security assessment is a well-established requirement and powerful analysis tools have already been proposed [5], including distributed architectures [6].
Some tools offer to define and automatically generate attacks, which then can be applied to a simulated communication network, including the actual code that then emulated for the different nodes [7, 8]. Some tools even provide their own descriptive language to specify the attacks carried out [9]. This allows to easily re-adjust the simulation whenever necessary and is also an important feature for our framework.
Additionally, some powerful discrete-event simulators enable researchers to write specialized tools for assessment of distributed communication in heterogeneous networks. Especially OMNeT++ is to mention for a variety of sub-frameworks as the INET framework and the ease in selecting the desired granularity of modelling [10, 11, 12].

Cited attack simulators lack the possibility to run the attacks decentralized, since some components are assumed to have a god-like view on all components. This prevents their use on actual Controller-Hardware-in-the-Loop setups.
Additionally, said tools simulate only a model of the sensor network, but not the actual implementation. This restricts the significance of the results. However, we propose is to include both possibilities, to define a model of the network but also to run the tests in a more realistic environment with minimal additional configuration.
Most tools use a centralized approach, that allows direct control of the simulator over each node. Then the simulated attacks are run by a central component of the simulator. This approach needing a centralized component is not always applicable on distributed control structures.
The OpenDISCO framework and prototype is novel to the extent that, on one hand, the control layer is also assumed to be decentralized, and on the other hand, the control algorithm itself is under investigation and its resilience properties are to be evaluated directly.
The importance of combined control- and power-hardware-in-the-loop approaches has recently received great attention also in the ERIGrid community [13], where a testing chain of Smart Grid control algorithms has been proposed. Here the researchers propose to gradually intensify the tests by introducing more and more realistic tests in their validated order. Our research complements this approach, since we aim to show that our solution can be easily integrated in the previously proposed testing chain, because it can be applied in software scenarios as well as part of control-hardware-in-the-loop scenarios.
The assessment of the system's behavior in an unreliable network has been studied with the focus of demand side management. Our research acknowledges the important work of Dambrauskas et al. at the Smart Grid Laboratory of Strathclyde University. We will utilize their method [14] to extend our scope, applying the emulator for realistic communication networks for the assessment of its impact on decentralized control of distributed energy generation (DER).
Finally, named attack simulation frameworks are not publicly available as open-source software, preventing in-depth usage of their features for further development. Our software will be made open-source.

4 Executed Tests and Experiments

We propose a framework that can **evaluate the security** of distributed control algorithms (DCAs) for the Smart Grid. At this step of our research, we are focusing on the **robustness** of said DCAs. One perspective goal is to describe the implications of different real-world communication links (e.g. landline DSL, GSM, LTE, 5G, PLC) on distributed control algorithms.

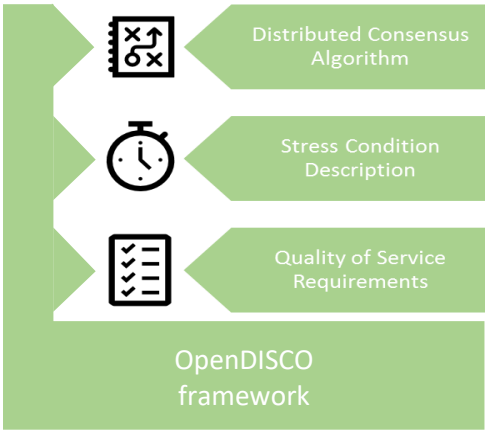


Figure 7: The OpenDISCO framework offers easy-to-use interfaces for developers and engineers to describe (a) the examined the DCA, (b) which stress conditions will be applied and (c) what QoS is required by specific application.

Framework Basic Structure

The proposed framework is a tool to help security researchers to assess an algorithm's robustness. Using a simple structure, it's possible

a) to describe a distributed control algorithm in C++,
b) to describe stress test conditions that will be applied to the network communication, and
c) to describe a quality of service (QoS) level that we be automatically compared against,

Using this configuration, the framework enables the

a) interaction with a cyber-physical system and the
b) communication between the nodes.

4.1 Test Plan

The goal of the proposed project is to illuminate **aspects of robustness** of Distributed Control Algorithms (DCAs) in decentralized Power Systems.

We want to show on a general level, that it's feasible to operate an event-driven message-based, extensible framework for distributed control of distributed energy resources (DER), which is in-itself an original approach in the field of real-time operation of secondary control.
Using this setup, we elaborate on distributed security assessment of DCAs. Our approach is to decentralize the control of the attacks to the designated control nodes.

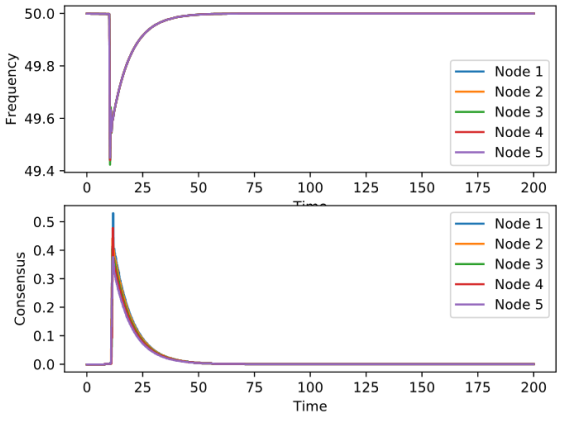


Figure 8: An example run of the proposed framework with the selected DCA for load-frequency stability. After a disturbance, in second 14 the nominal frequency of 50 Hz is restored within 30 seconds. Each line in the lower image represents the consensus computation of a single DER / control node.

To investigate this, our **experiment** comprises **runs** with increasing delays. Run 0 has no delay, run 1 has a delay of 500ms per message, run 2 has a delay of 1000ms and this pattern continues up to run 21 with a delay of 100.000 ms resp. 10 seconds. One **round** of the experiment consists of said 21 runs. We expect that with increasing delay, the occurrence of instabilities increases, such as overshoot, significant increase of time to recover the nominal frequency of 50 Hz, and finally the inability to recover 50 Hz at all.

We want to conduct as many rounds of the experiment as possible during the given lab access, to be able to statistically underline the found characteristics. Figure 8 shows an exemplary "run 0" without a delay, emphasising the expected behaviour of the DCA consensus and the CPS's response in terms of frequency over time.

4.2 Standards, Procedures, and Methodology

The main feature of our framework is the ability to assess DCAs that require **no central component**. We run experiments with a real-time-simulation of a **physical power system** with multiple Distributed Energy Resources (DER), which we call “control nodes” and of which each runs the exact same software. The task of the distributed control is to maintain the **frequency stability** of the overall power system via achieving a **distributed consensus** across all control nodes.
In our previous experiments we applied a sudden deviation in the power system’s frequency and had the distributed consensus algorithm **react to this disturbance**. Thereby we could show, that said distributed consensus algorithm **is able to recover the nominal frequency** of 50 Hz in well under 30 seconds. This we call the “no stress scenario”.
In our experiments, we applied different levels of stress to chosen DCA. These include different combinations of message delay, (temporary) node disconnect, multiple message re-transmission, message re-routing and jitter. During our TA at Strathclyde University, we aim to prove the reproducibility of our research using real-time digital simulation and Controller Hardware in the Loop.

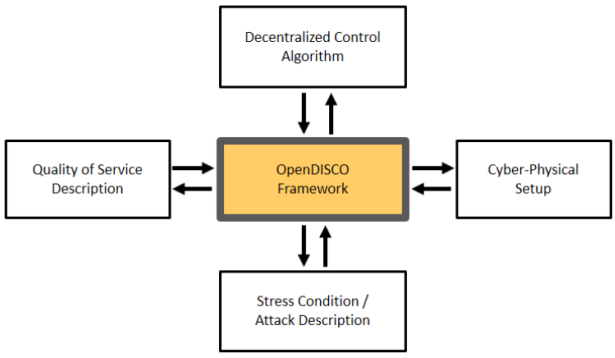


Figure 9: The joint function of the introduced framework.

4.3 Test Set-up(s)

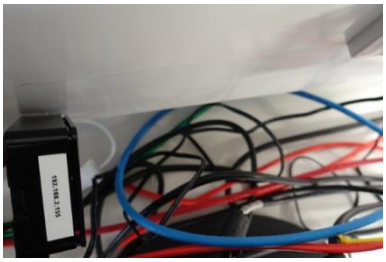
At the D-NAP, the pre-setup controller configuration consists of 5 Raspberry Pi embedded devices connected to an Ethernet LAN. The subnet is 192.168.2.0/24 and specifically the embedded devices have the IP addresses 192.168.2.151 to 192.168.2.155.

a) 192.168.2.151

b) 192.168.2.152

c) 192.168.2.153

d) 192.168.2.154



e) 192.168.2.155

Figure 10a-e: The used raspberry pi embedded devices as connected in the lab to the local network.

The investigated algorithm is a Distributed Consensus Averaging Droop-Control Algorithm. On each Raspberry Pi, the agent/algorithm receives data from the RTDS via IEC 61850 GOOSE messages. The received message contains the current frequency measurement at the agent's position in the simulated power system.
Additionally, all agents communicate over bilateral OPC UA connections and exchange their respective frequency measures.
After receiving a frequency measure from every other agent, the algorithm calculated the average of all measured frequencies. This average is sent as a control command to the respective power inverter, which is simulated at the RTDS, again via GOOSE message.

4.4 Data Management and Processing

The prepared run-script opens SSH connections to all five raspberry pi embedded computers and executes all necessary commands. Just log-in to any of the computers

```
ssh pi@192.168.2.151
cd ~/UHH/OpenDISCO-framework/scripts/
```

and type

```
bash start_rpis.sh -c "--topology fully"
or
bash start_rpis.sh -c "--topology fully -x ReceiveDelay500ms.xml"
```

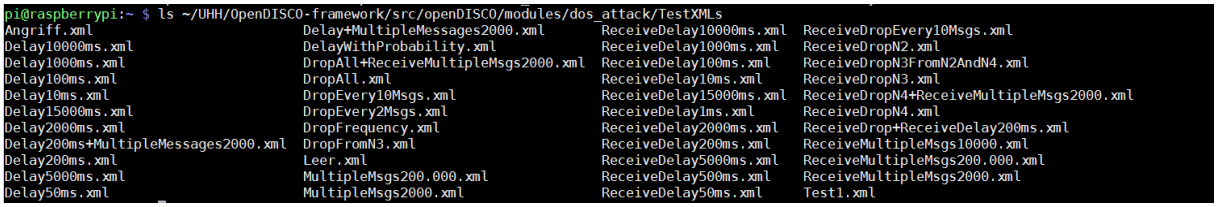


Figure 11: To apply any stress condition, start the affected agent with the addition parameter -x filename.xml

To see all available impairments / tree conditions, type

```
ls ~/UHH/OpenDISCO-framework/src/openDISCO/modules/dos_attack/TestXMLs
```

To see the output of the agent's executing, log in to any of the raspberry pi's and type

Agent 1

```
ssh pi@192.168.2.151
tmux a -t opendisco
```

to leave, type

```
CTRL-B  and then D
```

(for Detach)

Agent 2

```
ssh pi@192.168.2.152
tmux a -t opendisco
```

to leave, type

```
CTRL-B  and then D
```

(for Detach)

…

The frequency and consensus data are manually stored using the functionality of the RTDS control software.

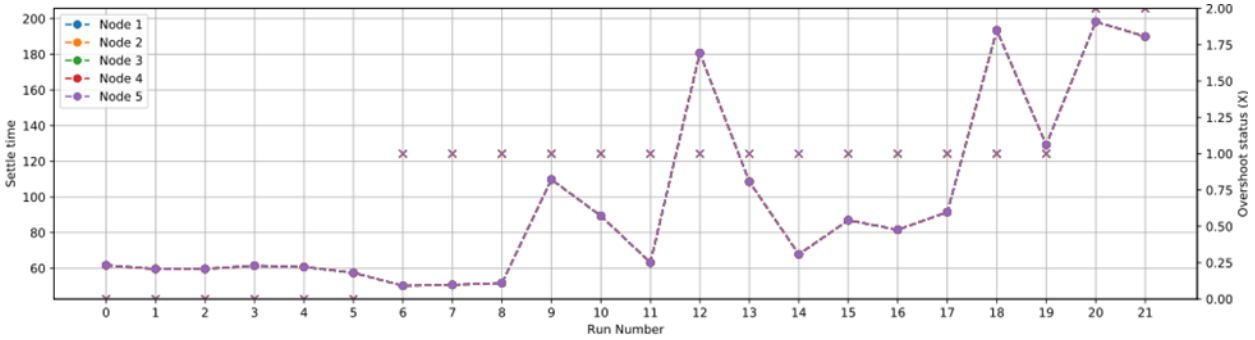


Figure 12: The settle time in one exemplary round of experiments.

Results and Conclusions

In our example implementation, we want to assess the security of an existing Primary Control algorithm. In Europe, the nominal grid frequency is 50 Hertz. When the frequency drops due to increasing load, the existing distributed algorithm takes action towards the nominal frequency by agreeing on a consensus value, in this case by averaging the suggestions of all five participating nodes, as you see in the figure, and then executing the consensus value as a primary control decision. We want to evaluate: Is the algorithm resilient against DOS attacks?

To investigate this, we conducted runs with increasing delays. Run 0 has no delay, run 1 has a delay of 500ms per message, run 2 has a delay of 1000ms and this pattern continues up to run 21 with a delay of 100.000 ms resp. 10 seconds.

One **round** of experiments consists of said 21 runs. Image 12 shows the average settle time of a selected round. In this specific round, the settle time of the algorithm for runs 0 to 5 is at around second 60, as illustrated by Figure 13 and Figure 14, showing run 0 with zero delay and run 5 with a delay of 2500 ms, where the single consensus steps are easily distinguishable.

Figure 12 also indicates for each run, whether an overshoot has happened, by marking the respective column with an x mark. This behaviour, as also seen in Figure 15, affects all runs from run 6 up to run 21. Our interpretation is, that the investigated distributed control algorithm is only resilient against overshoot for communication delays up to 2500 ms, meaning that if a controlled CPS has more strict requirements regarding overshoots, additional measures would be necessary.

Another finding of run 6 to 8 is, that the settle time of the DCA is unexpectedly decreasing instead of the expected increasing. This is explicable, since the delay causes the algorithm to use outdated data, implicating a stronger reaction to the (assumed) greater deviation from 50 Hz.

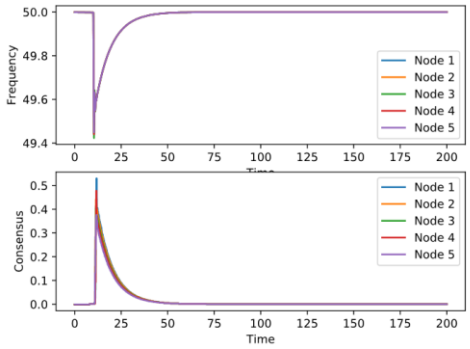


Figure 13: Run 0 of the experiment. No delay is applied, thus the control strategy shows the typical inversely proportional behaviour.

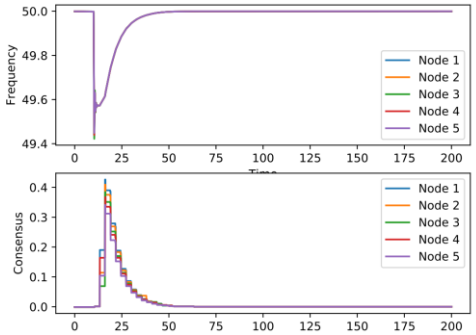


Figure 14: Run 5 of the experiment. The applied delay is 2500 ms. The delay between consensus steps is clearly visible.

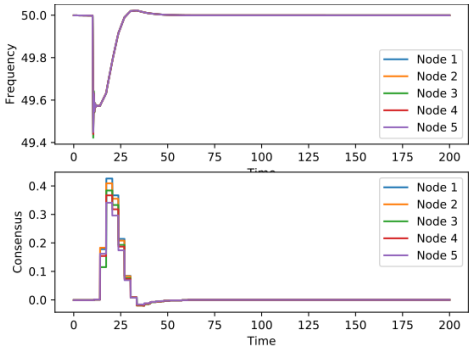


Figure 15: Run 6 of the experiment. The applied delay of 3000 ms causes an overshoot at around second 30.

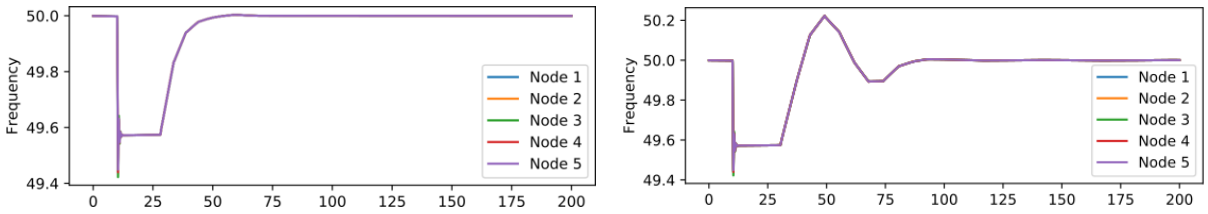


Figure 16a) Run 11 with a delay of 5000 ms *16b) Run 12 with a delay of 5500 ms*

This behaviour is indeed not always reproducible in all rounds of the experiment, since it depends of the exact timing, which is connected to the specific extent of the experienced frequency drop. This unreliable reproducibility explains the different outcome of run 11 and run 12, where the settle time is unpredictably different, although the applied delay is only off by 500 ms, or in total numbers: 5000 ms for run 11 respectively 5500 ms for run 12.
These two runs are illustrated in Figure 16 a and 16 b.

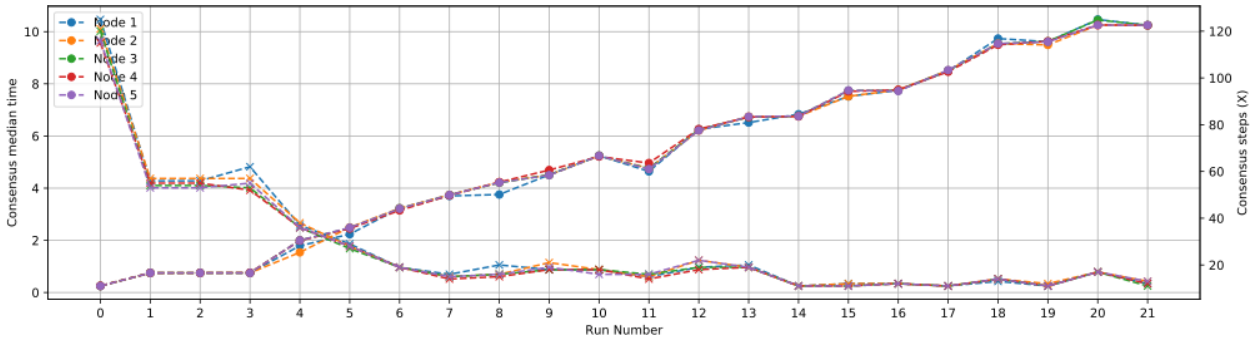


Figure 17: The number of steps necessary to reach a consensus (decreasing, marked as dots) versus the consensus median time (decreasing, marked as x)

Figure 17 shows the average number of steps that the consensus algorithm takes in each run to lead the frequency back to 50 Hz. This number is not representative, since the run-time of each run is limited to 200 seconds. The number of consensus steps is decreasing, seemingly unintuitive, caused by the effect, that the number of consensus operations is decreasing due to the delay. When the delay increases, e.g. up to 10 seconds, the number of possible consensus steps is by nature limited to 20 (which is the result of 200 divided by 10).

Figure 18 shows a possible effect of the delay attack, where each consensus is delayed to the point, where the result of the consensus step is actually counter-productive. This visualizes illustratively the need for a fall-back of this algorithm to ignore the consensus for the case, that local measurements contradict the global consensus decision.

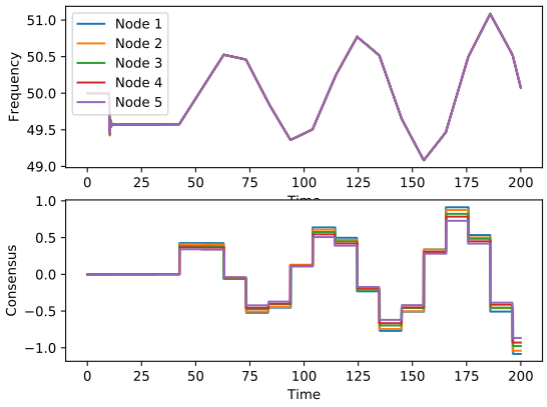


Figure 18: An example of run 20 of the proposed framework with an applied delay of 9500 milliseconds. The overshoot happens several times.

Open Issues and Suggestions for Improvements

The next steps should be to identify generalizable key strategies to improve DCA robustness. This comprises methods of attack detection on one hand (e.g. timeouts, thresholds, rule based alerting and anomaly detection), and methods of attack mitigation on the other hand (such as caching, anticipating of values, dropping of values and withdrawing of trust), as well as combined approaches such as defining limits for converging of results or median-based averaging, to mitigate tampering attempts or in case of inconsistencies between locally measured values and the consensus with other nodes.
Then, an adaptive approach could be elaborated, that would automatically determine the optimal strategy to react of denial-of-service attacks.
Our research included collecting data of the simulated attacks, to develop approaches for real-time detection of malicious behavior, mitigate techniques and ways to build up a reputation-based trust management. This data shall be used for future research.

Also, naturally occurring delays will be investigated, such as landline DSL, GSM, LTE, 5G, PLC, to evaluate the connectivity prerequisites of distributed energy system regarding different types of communication links. This is especially interesting when deciding where such a system of fully decentralized control can be realistically introduced. At the University of Strathclyde, recent research has shown that the Impact of realistic communications for fast-acting demand side management is not negligible [14]. Using a comparable setup and equivalent considerations regarding different types of connectivity, it could be shown how these impacts can be transferred to the domain of our experiment.
By simulating Quality of Service (QoS) properties, like latency and package loss, for connection types as cellular connections (GSM, LTE), land-line (DSL, ADSL+) and Power Line Communication (PLC), as depicted in Figure 17, a worthwhile research would be to elaborate on the pre-requisites necessary for this type of decentralized communication setup. We aim to describe the implications of each connection type regarding the evaluation of distributed control algorithms and propose best practices for selecting suitable algorithms for each communication type.

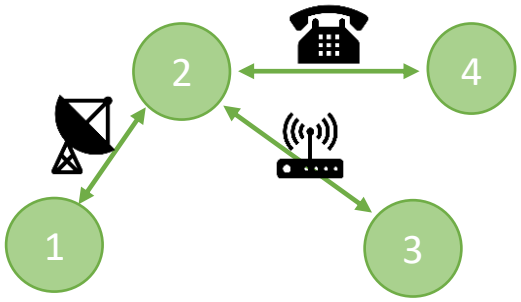


Figure 19: Simulation of different network links for different control nodes. The connection between node 1 and 2 is simulated to match the properties of a direct radio connection. Likewise the connection to node 3 represents a wifi connection and to node 4 an ADSL landline connection.

5 Dissemination Planning

The Smart Grid laboratory at Strathclyde University will gain experience with the OpenDISCO framework, enabling them to broaden their research methods. We are looking forward to a strengthened collaborational relationship and fruitful exchange of ideas and research methods, continuing after our stay at Strathclyde University.

Our research group plans to publish at least two papers out of the stay at Strathclyde university. Firstly, the operation of an event-driven message-based, extensible framework for distributed control of distributed energy resources is in-itself an original approach in the field of real-time operation of secondary control. With the results obtained from the experiments, we aim to publish a paper at the CIRED conference that takes place in June 2019 in Madrid, Spain.

We evaluated our pre-defined attack vectors, with the goal to gain better insight into distributed simulation of decentralized attacks on the Smart Grid and possible mitigation strategies. The gained results will be used to further evaluate possible enhancements to our existing distributed secondary control algorithm, hopefully documenting a direct improvement enabled by our framework and published in an article for the 8th DACH+ conference on Energy Informatics in October 2019 in Salzburg, Austria.

Currently, we are planning a follow-up research with the University of Strathclyde that aims to describe the effects of specific network configuration. The Smart Grid Lab of the University of Strathclyde provides us with a device to simulate the specific Quality of Service (QoS) properties, like latency and package loss, for Power Line Communication (PLC). We aim to describe the implications of this connection type regarding the evaluation of distributed control algorithms and propose best practices for selecting suitable algorithms for each communication type. Thereof we are planning a scientific publication for the Innovative Smart Grid Technologies ISGT 2020 conference.

Since the OpenDISCO framework is open-source and publicly accessible, each improvement implemented during our stay benefits the scientific community.

6 References

[1] Agent Based Distributed Control of Islanded Microgrid - Real-Time Cyber-Physical Implementation / Nguyen, Tung Lam; Tran, Quoc-Tuan; Caire, Raphael; Gavriluta, Catalin; van Nguyen, Hoa. In: Proceedings of the IEEE PES Innovative Smart Grid Technologies ISGT Europe 2017
[2] Open source as enabler for OPC UA in industrial automation. / Palm, F., Grüner, S., Pfrommer, J., Graube, M., & Urbas, L. (2015, September). In Emerging Technologies & Factory Automation (ETFA), 2015 IEEE 20th Conference on (pp. 1-6). IEEE.
[3] open62541 is an open source C (C99) implementation of OPC UA licensed under the Mozilla Public License v2.0. https://open62541.org/
[4] Lessmann, J., Janacik, P., Lachev, L., Orfanus, D.: Comparative study of wireless network simulators. In: 7th International Conference on Networking, ICN, pp. 517– 523. IEEE Computer Society (April 2008)
[5] Power system security assessment. / Morison, Kip and Wang, Lei and Kundur, Prabha. In IEEE Power and Energy Magazine, vol. 2, no 5, pp. 30-39, 2004, DOI: 10.1109/MPAE.2004.1338120
[6] A distributed architecture for online power systems security analysis. / DI SANTO, Michele, et al. In: IEEE Transactions on Industrial Electronics, 2004, 51. Jg., Nr. 6, S. 1238-1248.
[7] Y. T. Wang und R. Bagrodia. SenSec: A Scalable and Accurate Framework for Wireless Sensor Network Security Evaluation. In: 2011 31st International Conference on Distributed Computing Systems Workshops. Juni 2011, S. 230–239. DOI: 10.1109/ICDCSW.2011.26.
[8] Alvaro Diaz und Pablo Sanchez. Simulation of Attacks for Security in Wireless Sensor Network. In: Sensors 16.11 (2016). ISSN: 1424-8220. DOI: 10 . 3390 / s16111932. URL: http://www.mdpi.com/1424-8220/16/11/1932.
[9] G. Dini und M. Tiloca. ASF: An attack simulation framework for wireless sensor networks. In: 2012 IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). Okt. 2012, S. 203–210.
[10] Khan, A., Bilal, S.M., Othman, M.: A performance comparison of open source network simulators for wireless networks. In: IEEE International Conference on Control System, Computing and Engineering, ICCSCE, pp. 34–38. IEEE Computer Society (November 2012)
[11] Kumar, A., Kaushik, S., Sharma, R., Raj, P.: Simulators for wireless networks: A comparative study. In: International Conference on Computing Sciences, ICCS, pp. 338–342. IEEE Computer Society (September 2012)
[12] Leovigildo Sánchez-Casado u. a. NETA: Evaluating the Effects of NETwork Attacks. MANETs as a Case Study. In: Advances in Security of Information and Communication Networks. Hrsg. von Ali Ismail Awad, Aboul Ella Hassanien und Kensuke Baba. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, S. 1–10.
[13] Combined control and power hardware in-the-loop simulation for testing smart grid control algorithms. / M. Maniatopoulos, D. Lagos, P. Kotsampopoulos, N. Hatziargyriou. In: IET Generation, Transmission & Distribution, vol. 11, no. 12, pp. 3009-3018, Sept. 2017, doi: 10.1049/iet-gtd.2016.1341
[14] Impact of realistic communications for fast-acting demand side management. / *Dambrauskas, P.; Syed, M.H.; Blair, S.M.; Irvine, J.M.; Abdulhadi, I. F.; Burt, G.M.; Bondy, D.E.M.* In: Proceedings of 24th International Conference and Exhibition on Electricity Distribution. Stevenage, 2017.

7 Annex

7.1 List of Figures

7.2 List of Tables