pandaict: a Novel Open-Source Tool for modeling Cyber-Physical Energy Systems

Jan Ringelstein[1,*], Frank Marten[1], Mike Vogt[1], Gourab Banerjee[1], Johannes Bao[2]
[1] Department of Grid Planning and Grid Operation, Fraunhofer IEE, Kassel, Germany.
[2] Institut für Energiesysteme, Energieeffizienz und Energiewirtschaft, TU Dortmund, Germany.
[*] Corresponding author, email: jan.ringelstein@iee.fraunhofer.de

Abstract

As energy systems are digitized to maintain and improve security of supply with a high amount of fluctuating renewable generation, highly complex cyber-physical energy systems (CPES) arise. Holistic planning of these systems, including reliability and resilience analysis as well as investment decision support, needs tools for calculating and simulating energy and information networks and their interactions. The work at hand presents a novel Python-based open-source tool for modeling information networks called "pandaict". It is designed to be compatible with existing open-source tools for energy system modeling and is tested in a co-simulation in conjunction with those.

1 Introduction

1.1 Motivation

With the ongoing introduction of fluctuating renewable generation, controllable local flexibility and intelligent metering systems, energy distribution systems more and more transition into highly complex cyber-physical energy systems (CPES). These are characterized by the interplay between classic SCADA and control and supervision of assets in medium- and low voltage (MV, LV) networks. The latter include local measurements and controllable systems down to the household level, e.g. electric vehicle (EV) charging stations. Communication relies on infrastructure of distribution system operators, but also uses secure channels over public information and communication technology (ICT) networks. Multiple market players, such as grid and metering system operators, internet service providers, or e-mobility fleet operators contribute to or access the system.

A general goal of CPES is to enhance the security of supply and maximize the use of renewable sources. Functions vital to this include grid-friendly usage and dispatch of flexibility of load and distributed generation, online grid supervision, and advanced fault detection and mitigation. ICT and energy networks are mutually interdependent, thus need to be considered as a compound system. This stresses the need for modeling and simulation tools that holistically enable CPES reliability and resilience improvements, scenario-based system planning, and investment decisions.

Properties, behaviour and mutual influence of electric grid, ICT network, controllable and non-controllable load and generation are typically subject to simulation-based assessment in current or future scenarios. Since oftentimes a single tool is not sufficient for this, multiple interacting tools need to be used. If those can not be natively integrated, they need to be bundled by means of co-simulation, which again needs a support framework that in itself represents another item in the toolset. Finding and combining tools that fit the use case at hand is the basic problem in CPES simulation.

1.2 State of the Art

CPES and cyber security simulation has been a contemporary smart grid research topic since well over a decade. Previous work has used a variety of different tools for ICT, electric network simulation and optimisation & control [1]. For tool coupling, there is a variety of co-simulation frameworks as assessed by Vogt et al. [2]. NS-2, NS-3, OPNET, and OmNet++ are popular ICT network simulators that may be used for smart grid simulation as assessed by Aslam et al. [3]. In terms of tool application, usually specific use cases are considered, e.g. load shedding, voltage control, state estimation, energy management, system restoration or other control involving ICT. Cyber security and vulnerability assessment is currently a focus, as CPES are recognized as critical infrastructure [4][5]. A review of smart grid cyber security requirements, attack vectors and vulnerabilities as well as countermeasures is provided by Achaal et al. [6].

Arguing that open-source tools are needed to support coupled ICT- and energy grid simulations in large-scale scenarios, the research project ReCoDe ("Reference platform Co-Simulation Digital energy systems") was started in 2022. This project specifically aims at bundling several existing tools and especially couples them with the ICT system simulation tool *pandaict* which is further developed in this project. The following sections shortly introduce the tools integrated. Section 2 describes main pandaict concepts and functions. Section 3 contains results from a first application-oriented proof-of-concept, while section 4 concludes the paper with an outlook.

1.2.1 pandapower

The open-source power system simulation tool *pandapower*[1] [7] combines the data analysis library Pandas

[1] http://www.pandapower.org

and the power flow solver PYPOWER for electric grid calculations. It is a joint development of the department e^{2}n at the University of Kassel and the Division of Grid Planning and Grid Operation at Fraunhofer IEE, Kassel. It enables a high degree of automated analysis and optimization of electrical power systems.
Several functionalities for power system analysis, i.e., power flow, state estimation, short-circuit calculation, optimization, controller simulation, topology searches, contingency analysis, time-series simulation, power system protection, etc. are offered by pandapower. Different from many other open-source network analysis tools, pandapower uses an element-based model for electric grids, where models for electrical components such as transformers and lines are connected to buses instead of using generic branches to connect the buses (bus/branch model). It provides the facility to calculate and simulate the electrical grid for all voltage levels from low to extra-high voltage, which allows power system studies for distribution and transmission grids.

1.2.2 SIMONA

The energy system simulation *SIMONA* was developed to analyze the behavior of smart grids in planning processes incorporating new technologies. It can derive realistic time series of all relevant variables in distributions grids based on a bottom-up approach with detailed asset models.
SIMONA uses a multi-agent-system with the network users as agents (system participant agents (SPA)). This approach allows parallel computing which is particularly useful for large simulation scenarios. The SPAs include physical models for the calculation of generated and consumed power [8][9].
SPAs can represent entities such as PV plants, household loads, and electrical storage systems. In order to model smart grid behavior, SPAs can be controlled by energy management agents (EMA). These EMAs are responsible for aggregating flexibility options, calculating set points and communication control signals to the SPAs. The set point calculation is based on individual objectives, such as self-optimization or achieving a target power output [10].

1.2.3 OpSim

OpSim[2] [11] is a co-simulation environment created by Fraunhofer IEE and University of Kassel. Its creation was motivated by the fact that real energy systems are far too complex to be modeled with one single simulator. Such systems contain multiple actors (e.g. distribution and transmission system operators), different technologies, and energy conversion between different grids (e.g. Power2Heat, Power2Gas). OpSim can couple different simulators and controllers to more accurately model complex multi-actor and multi-energy systems. Its function is illustrated below (**Figure 1**). It consists of a flexible message bus (RabbitMQ) via which so called “clients” exchange messages. The code and function of each client is the same, they are generic. Behind each client, a “proxy” translates incoming and outgoing information to/from a specific simulator (e.g. pandapower, SIMONA) or controller. Hence, proxies are specific interfaces. This architecture enables a high flexibility: to change one simulator, only one proxy has to be adjusted. Synchronisation between clients and organization of data exchange is handled by the MCP Core. A graphic user interface (MCP GUI) allows for scenario upload and simulation control.

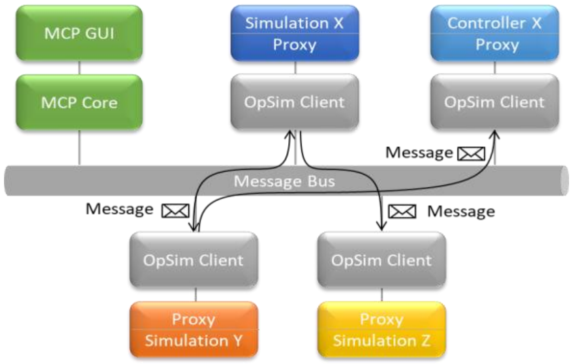

Figure 1: Overview of OpSim environment

2 Concepts and features of pandaict

The network simulators mentioned in section 1.2, like e.g. OmNet++, typically allow in-depth modeling of physical data flows in ICT networks, enabling investigation of byte-wise propagation of data or data packages using specific communication protocols within the physical network. In contrast to that, the aim of pandaict is to provide a more abstract and easier to use modelling tool that can calculate average statistical values and reliability metrics, and is directly compatible to pandapower. Just like the other tools in the “panda family”, it is provided as a Python library that defines extendable data structures and functions.
Currently, pandaict is in its alpha development state, i.e. functions are under test and not yet fully validated. Still, we encourage researchers and users to consider the tool for their use case, as our aim is to further develop it as open-source, keeping it very close to application. The current release may be obtained from the corresponding author.

2.1 Core and Connectors

The idea of pandaict is to allow for modelling stand-alone ICT networks (e.g. for telecommunication) as well as couple these with supply network models for e.g. electricity. Hence, the tool is split into *core* and *connectors*. The core is always needed and can be used stand-alone for generic ICT system modelling. The connectors can optionally be used and establish the coupling to other tools. Currently, there is only one pandapower connector available. It extends the core model by electric network elements, e.g. mapping of ICT devices to generators or loads.

2.2 Data Model

A pandaict network model involves three layers: the physical layer, the application layer and the information layer.
The physical layer contains three types of elements: *Nodes*, *Devices* and *Links*. They respectively represent physical

[2] http://www.opsim.net

network interconnection points (e.g. routers), physical end devices (e.g. meters), and physical interconnections between them.
For the latter, pre-defined standard link types (e.g. Gigabit Ethernet) provide data bandwidth, latency and reliability metrics for the links. Links can be bidirectional or unidirectional. They have upstream and downstream directions to represent asymmetric transmission technologies. Router-type nodes may connect network parts with different link types. Links may interconnect nodes or devices and nodes, but two devices can not be linked without at least one node in between. Thus, a minimal network contains two devices and one node (**Figure 2**).

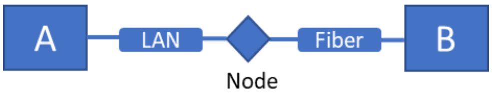

Figure 2: Two devices (A,B) connected by a minimal network with a router-type Node

At the application layer, *Apps* represent services that act as logical communication endpoints. One or more apps can run on the same physical device. *Connections* between apps represent end-to-end interconnections using ICT protocols for transmission of information items (e.g. a power measurement). Standard protocol types (e.g. IEC-61850) define information items supported by protocols as well as data volumes for them (**Figure 3**).

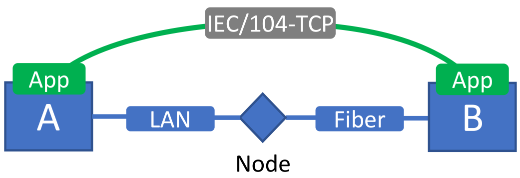

Figure 3: Connection with protocol between apps running on devices A and B

A special *Converter* element is used to model protocol conversion (**Figure 4**). A converter is technically a device that hosts a protocol conversion app. Just as the links, connections have an upstream and downstream direction.

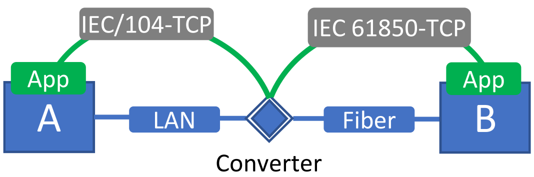

Figure 4: Usage of a protocol Converter device

Finally, the information layer defines the information items that are transmitted over the connections (**Figure 5**). The pandaict core defines only generic information types (e.g. REQUEST or URL) while the pandapower connector defines types relevant to the power network (e.g. ACTIVE_POWER for a power measurement). For each protocol type, supported information items and the payload needed to transmit them are defined. Definition of average sending rates for information items is possible, too.

All type definitions may be extended by the user, such that using custom link types, protocols or information items is possible.

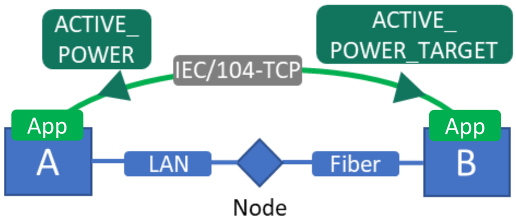

Figure 5: Information elements transmitted using a connection

2.3 Routing and Topology

The simple building blocks defined in the previous section can be used to model arbitrarily complex ICT systems in an abstract way. Here for, the pandaict advanced programming interface (API) provides basic functions including creation, storage, visualization and consistency checking. With the full model, different calculations can be conducted, mostly based on graph algorithms.
One of the most basic calculations is to determine if an end device is reachable from another, i.e. if there is a valid path through the network. This is obtained by a simple path search in the physical layer for the connections defined in the application layer, e.g. using the Djikstra algorithm. Here for, the physical layer is internally translated into a graph which is analysed using the NetworkX Python library. Furthermore, routing functions are provided that rely on the definition of a routing algorithm. In the current pandaict version, only shortest path / lowest latency routing is included, but new routing algorithms may be added by the user. Algorithms that use multiple routes, e.g. random routing, as well as different routes for up- and downstream directions are supported. Here for, so-called “link usage probability” (LUP) tables are calculated for each connection, which store possible routes that information items can take through the network. This is done for either direction. Links, hops and usage probabilities are calculated for each route according to the routing algorithm. Combined with information item, average sending frequency and payload definitions, this enables calculation of mean link bandwidth usage, end-to-end data volume and total latency at connection level for both up- and downstream directions, respecting the routing as possible given the current state of the physical network. Since this state can be easily changed, e.g. by putting physical elements out of operation, investigation of ICT element outage scenarios as part of a resilience analysis becomes possible.

2.4 IP routing support

The current version of pandaict features a general support for IPv4 networks. IP addresses can be defined for devices and nodes, and routing tables with support of static and dynamic routing can be calculated. This function is however currently limited, as definition of subnetworking is not fully supported and IPv6 support is not yet implemented.
Special support of radio and internet-based connections is currently not available in pandaict. The same is the case for

connections that use virtual private networks (VPN), i.e. secured communication over shared physical media. Such connections may be modeled in a simplified way, though. The simplest option for this would be using links with adapted parameters, e.g. a latency typical for internet or VPN, or a limited bandwidth that reflects shared usage of a physical medium, e.g. the cellphone network.

2.5 Reliability assessment

Communication reliability assessment is one of the primary pandaict use cases. Generally, all ICT physical elements are considered repairable. Thus, the pandaict core natively supports mean time between failure (MTBF), mean time to repair (MTTR) and bit error rate (BER) as basic reliability metrics in the physical layer. **Table 1** shows which elements carry the metrics.

Table 1: Reliability metrics by element type

Element	MTBF	MTTR	BER
Node	X	X	X
Device	X	X	
Link	X	X	X

For the links, the BER is given per kilometer by the link-type, assuming that bit errors may occur by crosstalk or external interference. This value is multiplied with the link length to obtain the total BER. The availability A, which is another commonly used metric, is calculated as:

$$A = \frac{MTBF - MTTR}{MTBF}$$

The problem is now to calculate the same metrics at the connection level, i.e. for information elements that are transmitted over multiple possible routes, using various physical layer elements that are arranged in a system [12]. A single connection-level system is represented by an LUP table and generally consists of r parallel routes, each representing a serial system (**Figure 6**).

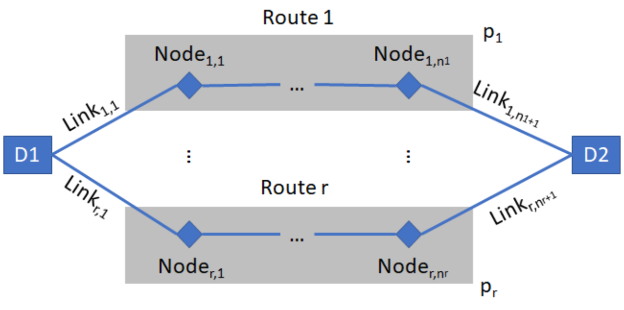

Figure 6: Physical layer reliability system diagram for connection between apps running on devices D1 and D2

The reliability metric calculation for each single route relies on iteratively combining the current metric values with the next element along the path from sender (D1) to receiver (D2). When it comes to total system reliability however, there are several fallacies: on one side, different routes might partially use the same nodes and/or links. On the other side, there are routing algorithms (e.g. flooding) where information items are copied and transmitted over multiple routes, such that route usage probabilities $p_1 ... p_r$ may add up to a value greater than 1. Furthermore, different connections might use the same link or node elements too, leading to multiple connection outages when such elements fail. This is a typical case in practice but makes reliability metric calculation a non-trivial problem. As simplification, the current implementation assumes that information items are not copied within the network, thus each item uses exactly one of the possible routes. This is the case for e.g. shortest-path routing algorithms and makes it possible to calculate the total connection system reliability by adding up the route reliabilities using the probabilities as weights. The iterative MTBF calculation is thus done as follows. First, the MTBF is calculated for each route. In the first iteration i=1, the initial value is set as combined MTBF of the first two elements in route j:

$$MTBF_j(1) = \frac{MTBF_{j,1}^{link}\ MTBF_{j,1}^{node}}{MTBF_{j,1}^{link} +\ MTBF_{j,1}^{node}}$$

For each coming iteration $i = [2..(n_j + 1)]$ where n_j is the number of nodes in route j, MTBFs are subsequentially added using a helper value $MTBF_H$:

$$MTBF_H = \frac{MTBF_j(i-1)\ MTBF_{j,i}^{link}}{MTBF_j(i-1) + MTBF_{j,i}^{link}}$$

$$MTBF_j(i) = \frac{MTBF_H\ MTBF_{j,i}^{node}}{MTBF_H + MTBF_{j,i}^{node}}$$

For the last iteration, $MTBF_H$ is already the final result since there is no more node to add. The values per each route thus obtained are used to calculate the total for the connection:

$$MTBF^{total} = \sum_{j=1}^{r} p_j\, MTBF_j$$

The MTTR calculation is done in a similar way. Finally, the BER for route j is calculated as:

$$BER_j = 1 - \left(\prod_{n=1}^{n_j} \left(1 - BER_{j,n}^{node}\right) \prod_{l=1}^{n_j+1} \left(1 - BER_{j,l}^{link}\right) \right)$$

And the total BER is:

$$BER^{total} = \sum_{j=1}^{r} p_j\, BER_j$$

Note that this needs to be done separately for upstream and downstream directions of the connections since they may use different routes, links and nodes.
Also, the problem of multiple node and link usage and multiple connection outages remains and needs to be carefully considered during the modeling in order to obtain meaningful reliability metrics. An example for this is described in the next chapter. Still, this poses a limitation to be worked out in coming releases.

3 Application example

3.1 Use Case

As mentioned in section 1, the overall goal of project ReCoDe is to bundle different tools – in this case, pandaict, pandapower, SIMONA and OpSim – to allow holistic analysis of CPES (**Figure 7**). Hence, a simple use case was designed as a proof-of-concept.

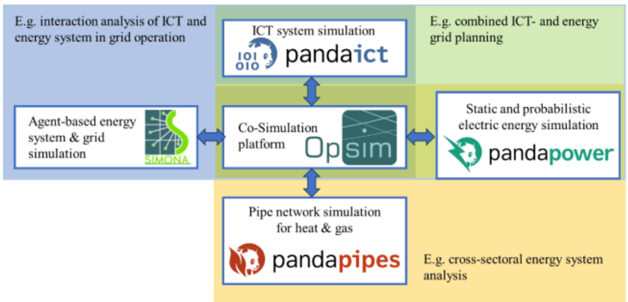

Figure 7: Example setup of pandaict in a combined CPES co-simulation

The use case defines a MV/LV distribution network with EV charging stations that are used as flexibilities with the goal to lower grid load and minimize cost for grid reinforcements. The intended use of pandaict is to estimate how ICT network reliability and robustness affects electric network operation and reinforcement cost estimation. This requires estimation of ICT communication outage probabilities and durations for connections between the grid control center, grid measurements and EV charging stations within a one-year simulation timeframe.

3.2 Electrical scenario grid and co-simulation setup

As electric network scenario, an LV semi-urban grid from the Simbench dataset was used [13]. It includes one transformer station and five feeders with a total of 112 buses, 15 of which are supplying a household with controllable charging station (**Figure 8**). Time series for the non-controllable loads are also provided by Simbench. Generators are not explicitly modeled. It is assumed that there is a central grid control station which has full information about bus voltages, line and transformer loadings in the network, and can control the active powers of the households with charging station.
The main co-simulation implements a quasi-stationary timeseries simulation for one year. It consists of three components: (i) grid calculation for performing load flow calculations and determining the grid state in terms of voltages and loadings for all buses, lines and transformers (ii) grid operation which checks the grid state for limit violations and (iii) simulation of households with controllable active power in order to lower the grid loading. Components (i) and (ii) are implemented as OpSim proxies which use pandapower for the load flow calculation.
The households (iii) are implemented as system participant agents in SIMONA. In between the components, messages carrying grid measurement values, active power setpoints and active power actual values are exchanged through the OpSim message bus.

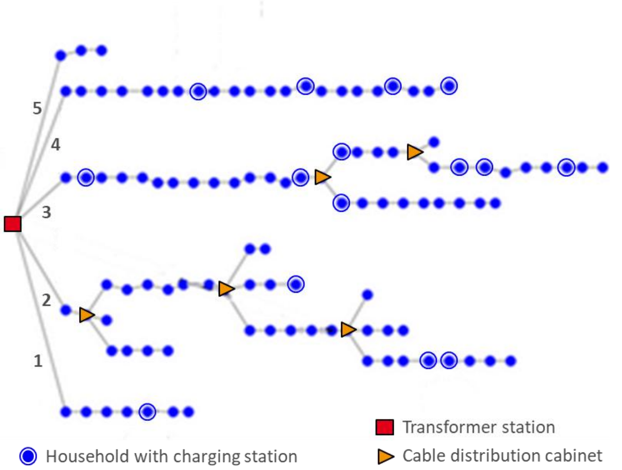

Figure 8: Electric network topology for the proof-of-concept

3.3 ICT network

While the base use case setup assumes that the co-simulation subcomponents can communicate without disturbance, pandaict introduces connections with limited availability. This relates both to the grid measurements sent from grid calculation to grid operation component, and active power setpoints sent from the grid operation component to SIMONA. Actual power values are sent from SIMONA to the grid calculation, but are assumed to have 100% availability since they do not rely on physical data transmission in the real system.
An ICT network was designed on top of the electric network, assuming ICT nodes at the control center, the trafo station, and the cable distribution cabinets. Each of the latter nodes is connected to measurement devices that capture loadings and bus voltages of the grid area beneath, and forward them to the respective higher-level node in the chain, creating a node hierarchy that follows the grid topology and along which measurement data is bundled. For modeling this, converters were used instead of nodes. This is a workaround for the multiple link usage / connection outage problem mentioned in section 2, as it allows for separate calculation of reliability for connections that use the same links and/or nodes. With this, simultaneous outages of multiple measurements can be represented.
In terms of the charging stations, the control center is also connected to the households. The ICT model assumes connection over secure public internet and intelligent metering infrastructure here.

3.4 Interfaces

The interfaces between the co-simulation components are realized through the OpSim message bus and data model, which directly supports all measurements and setpoints communicated. The latter items are represented by so-called “OpSim assets”. The interface to pandaict however is based on calculation of reliability metrics prior to the timeseries co-simulation. For this, there is a Python script that uses the pandaict core and pandapower connector to setup the ICT model and calculate MTBF and MTTR metrics for all connections between control center, measurement devices, households and intermediate converters. The result is represented as an NXGraph (**Figure 9**) and remains constant unless the ICT network is changed.

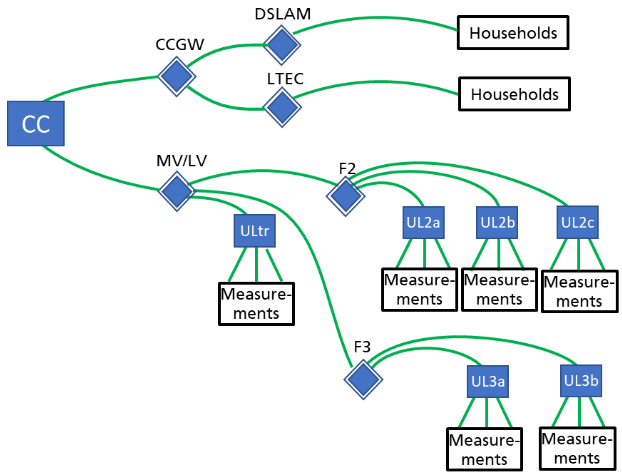

Figure 9: NXGraph representing connections and ICT network elements. CC: Control Center, CCGW: CC Gateway, DSLAM: Digital Subscriber Line Access Multiplexer, LTEC: Long-Term Evolution (4G) base station, MV/LV: trafo station, F2/F3: Cable distribution cabinets, ULx: measurement devices.

A string representation of this graph is distributed to all OpSim simulation components as part of the OpSim scenario. The grid calculation and grid operation utilize the reliability information through a connectivity generator script that calculates random outage times and durations for measurements and setpoints during the timeseries simulation, according to the MTBF and MTTR values. The connectivity generator features a function that can directly output if OpSim assets are reachable by the control center at any given time within the simulation timeframe. With this, it is possible to perform Monte-Carlo simulations to judge robustness of grid operation for random ICT outages, as similar to previous work [14]. The result can also be made deterministic by fixing the random number generator seed. To judge how changing the ICT components affects grid operation, repeated simulations with different ICT network configurations and parameters may be carried out.

3.5 Proof-of-concept

To provide a proof-of-concept, a fixed random seed was selected. The grid operation component was configured to detect if the transformer loading would exceed a threshold in each time step. If so, an active power setpoint of zero valid for the coming hour would be sent to the households, indicating they should not source or feed power from or into the grid. The intention of this was to reduce the grid loading presumably caused by too high load.
A first result was obtained using a simplified replacement for the SIMONA component, with the main aim to test if all interfaces were operational and the overall system was working as intended. The households were configured in a highly simplified manner for this. A constant base load of 500 W per household was chosen. An 11 kW EV charging power was added to this for a consecutive period of 5 hours with starting time between 16:00 and 19:30, fixed depending on the household index. This resulted in a ramp-up and ramp-down of EV charging at the end of each day and caused frequent transformer loadings over 20%. The latter value was used as threshold for creating the deration signal, which was followed by the simplified households directly by shutting down active power to zero. However, in case of an ICT outage, the signal either could not reach one or more households, or would not be generated by the grid operation component at all due to unavailable grid measurements. **Figure 10** shows the transformer loading for a selected 24-hour period from the simulation result data. The grey area represents the time of a major ICT outage which rendered all grid measurements unavailable. Two simulation runs were executed with this: a reference without ICT outage, and an outage case. In figure 10 the reference is represented by the solid line, and the outage case by the dotted line. Note that the loading needs to exceed the 20% threshold for at least one 15-minute timestep before the deration signal is sent. It is apparent that the loading could be reduced by the deration signal only if no outage was present, which matched expectation.

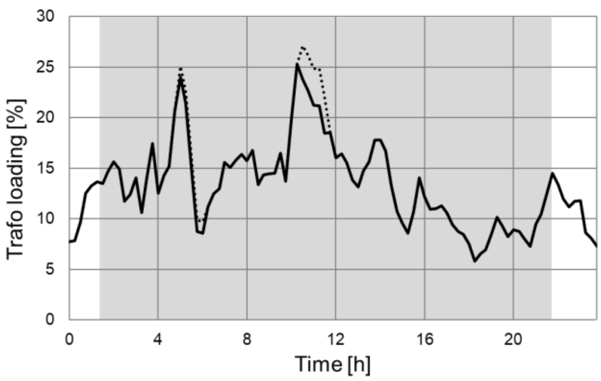

Figure 10: Simulated effect of ICT outage with simplified household component

Following this successful first test, a full co-simulation was conducted, bundling all sub-simulation parts. For this, comparatively complex SIMONA household models were used. They include PV-storage hybrid systems with an energy management system that can dispatch power such as to reduce the total power towards zero if the deration signal was received. For this simulation, the threshold for signal generation was chosen to 15% transformer loading. As can be seen in **Figure 11**, which again shows a selected 24-hour period with a major ICT outage, differences between the outage and reference cases appear. However, in this case the transformer loading is lower in the outage case, which contradicts the expectation and intention of the deration signal. A closer look at the model behavior reveals that this is caused by each household feeding in about 10 kW of PV active power if the signal is not received. This is correct because in this case, households may consume or feed-in arbitrary active power, but it here leads to the lower transformer loading. Hence, the control mechanism chosen is not adequate to reduce grid loading when households feature distributed generation. Here for the control would need adaptation, but this is nevertheless out of scope of this proof-of-concept. Overall, it was successfully demonstrated that the tool coupling and ICT outage simulation is operable.

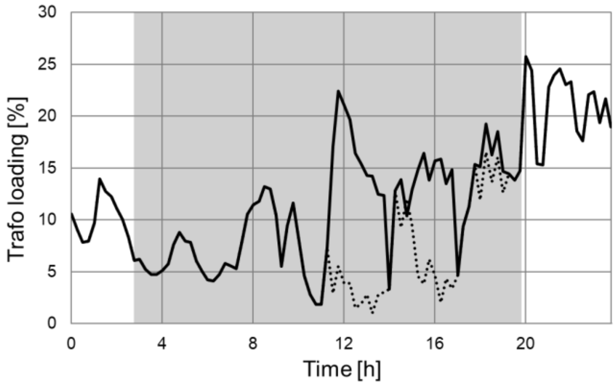

Figure 11: Simulated effect of ICT outage in full co-simulation

4 Conclusion and Outlook

The current alpha version of pandaict is a tool for modelling ICT networks that connects to the electric network calculation software pandapower. Both tools are characterized by high abstraction and ease-of-use. API documentation is provided using Sphinx. The usage of pandaict in a CPES co-simulation involving a tool coupling between OpSim and SIMONA was successfully demonstrated by a proof-of-concept.

The feature selection for coming releases of pandaict will be application driven. Currently, several features are planned for further developments. This includes improving support of IPv4 subnetworking and IPv6 support. Total system reliability and multiple connection outage calculations are also to be improved. To improve resilience analysis, an automatic identification of critical ICT network components is considered. Also, the creation of ICT networks is currently possible through the API only. This may be improved by defining an Excel-based storage format that can be edited manually or be automatically pre-filled with information from the electric grid model, in case pandapower is used alongside.

The pandaict calculation results need to be validated in terms of correctness. This may be done by manual reference calculations, which is however non-trivial even for ICT networks of moderate complexity. Using other network simulation tools calculating the same results may provide a validation reference. Finally, parameters for the standard ICT elements, namely links and information item data package sizes per protocol, are currently only roughly estimated. To improve this, they may be researched, taken from other tools or measured in a test setup, which would be possible at the control center laboratory of Fraunhofer IEE.

The current pandaict release may be obtained by directly contacting the corresponding author.

5 Acknowledgements

This work presents results from the project ReCoDe funded by the German Federal Ministry for Economic Affairs and Climate Action under grant no. “03EI6093”. The authors are solely responsible for the contents of this publication.

6 Literature

[1] K. Mets, J. A. Ojea, and C. Develder, “Combining Power and Communication Network Simulation for Cost-Effective Smart Grid Analysis”, IEEE Communications Surveys& Tutorials, Vol. 16, No. 3, 2014

[2] M. Vogt, F. Marten, and M. Braun, „A survey and statistical analysis of smart grid co-simulation”, Applied Energy 22, 2018.

[3] M. M. Aslam, W. Li, W. Liu, Y. Qi, U. Saleem, S. Riaz, “A review of integrated modeling and simulation of control and communication systems in Smart Grid”, Computers and Electrical Engineering 119, 2024

[4] J. Haack, A. Narayan, A.D. Patil, M. Klaes, M. Braun, S. Lehnhoff, H. de Meer, and C. Rehtanz: A Hybrid Model for Analysing Disturbance Propagation in Cyber-Physic al Energy Systems, Electric Power Systems Research 2022, Vol. 212, doi: 10.1016/j.epsr.2022.108356

[5] A. Narayan, M. Brand, and S. Lehnhoff: Quantifying the resilience of ICT-enabled grid services in cyber-physical energy system, Energy Informatics 2023, 6(Suppl 1):23, doi: 10.1186/s42162-023-00287-y

[6] B. Achaal, M. Adda, M. Verger, H. Ibrahim, and A. Awde: “Study of smart grid cyber-security, examining architectures, communication networks, cyber-attacks, countermeasure techniques, and challenges”, Cybersecurity 7:10, 2024

[7] R. Bolgaryn, G. Banerjee, S. Meinecke, H. Maschke, F. Marten, M. Richter, Z. Liu, P. Lytaev, B. Alfakhouri, J. Kisse, and D. Lohmeier: Open Source Simulation Software pandapower and pandapipes: Recent Developments, 2023, Open Source Modelling and Simulation of Energy Systems, doi: 10.1109/OSMSES58477.2023.10089685

[8] C. Kittl: Entwurf und Validierung eines individualitätszentrierten, interdisziplinären Energiesystemsimulators basierend auf ereignisdiskreter Simulation und Agententheorie, Düren: Shaker Verlag, 2022 [Online], doi: 10.17877/DE290R-22548

[9] J. Hiry: Agent-based discrete-event simulation environment for electric power distribution system analysis, Düren: Shaker Verlag, 2022 [Online], doi: 10.17877/DE290R-22549

[10] D. Feismann, S. Peter, J. Bao, T. Oberließen, and C. Rehtanz: Determining the influence of energy management systems on simultaneity factors of distribution grids, CIRED 2024 Vienna Workshop (CIRED 2024)

[11] OpSim: test- and simulation-environment for grid control and aggregation strategies, " 2024. [Online]. Available: www.opsim.net/en.

[12] The Safety and Reliability Society: Applied R&M Manual for Defence Systems, Part D - Supporting Theory, Chapter 6 - probabilistic R&M parameters and redundancy calculations. [Online]. Available: https://sars.org.uk/BOK/Applied%20R&M%20Manual%20for%20Defence%20Systems%20(GR-77)/p4c06.pdf

[13] S. Meinecke, D. Sarajilić, S. R. Drauz, A. Klettke, L.-P. Lauven, C. Rehtanz, A. Moser, and M. Braun: SimBench—A Benchmark Dataset of Electric Power Systems to Compare Innovative Solutions Based on Power Flow Analysis. Energies 2020, 13, 3290, doi: 10.3390/en13123290

[14] M. Panteli and D. S. Kirschen: Assessing the Effect of Failures in the Information and Communication Infrastructure on Power System Reliability, in proceedings IEEE PES Power Systems Conference and Exposition, 2011, doi: 10.1109/PSCE.2011.5772565