Final Technical Report (FTR)

a. Federal Agency	Department of Energy	
b. Award Number	DE-EE00031266	
c. Project Title	CyDER: A Cyber Physical Co-simulation Platform for Distributed Energy Resources in Smartgrids	
d. Principal Investigator	Christoph Gehbauer Senior Scientific Engineering Associate cgehbauer@lbl.gov 510-486-4146	
e. Business Contact	Brandy Jones Phone: 510-486-6018 Email: BJones@lbl.gov	
f. Submission Date	07/31/2019	
g. DUNS Number	078576738	
h. Recipient Organization	LBNL	
i. Project Period	**Start:** 3/3/2016	**End:** 7/31/2019
j. Submitting Official Signature		

Acknowledgement

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Solar Energies Technology Office (SETO) SunLamp Program Award Number 31266.

1. Executive Summary

The CyDER project aimed at developing an open-source, modular and scalable co-simulation platform for power grids with large shares of Distributed Energy Resources (DERs). The project partners are the Lawrence Berkeley National Lab (LBNL), Lawrence Livermore National Lab (LLNL), PG&E, SolarCity, and ChargePoint. The prime recipient is LBNL; SolarCity and ChargePoint were partners for the project's first two years.

Increased DER integration introduces a number of challenges in power grid operation including a more dynamic interaction between the transmission grid and distribution grids, and increased modeling complexity. Although specialized software exists to precisely model different components of the power system, it is far from trivial to integrate all various models and perform a holistic simulation. Instead of replicating all models in a common simulation program, a commonly accepted approach to tackle this model diversity is to couple third-party simulators and models through a co-simulation platform that coordinates information exchange among the various components.

Following this line of research, this project's objective was to develop a co-simulation platform based on a widely accepted industrial standard called Functional Mockup Interface (FMI). Within this process, the project developed models compliant with the FMI standard, called Functional Mockup Units (FMUs), and used them to perform various operational and planning power system analyses. Relying and building upon an industrial standard is the main differentiation of this project compared with previous or parallel efforts in the co-simulation area. Particular emphasis was put on delivering software utilities to facilitate setting up and running co-simulations by end-users. Furthermore, a strong aspect of this project is demonstrating that co-simulation techniques can be used to perform Hardware-in-the-Loop (HIL) simulations that couple software components (e.g., simulated models) with hardware components (e.g., real devices such PV systems and batteries). The long-term goal of CyDER project is to help establish FMI as a powerful standard for co-simulation and promote adoption by electric utilities and other interested stakeholders.

The main accomplishments of the project include the development of several FMUs including distribution and transmission grid models, PV inverters with Volt/Var/Watt controllers, batteries, and predictive optimal controllers. Additionally, a unique software package was developed, called SimulatorToFMU, which is capable of exporting any Python-driven simulator or Python script as an FMU. This is an important contribution towards establishing FMI as one of the main co-simulation standards, because more and more third-party programs for sub-system modeling and simulation are delivered with Python APIs. The CyDER platform was used to perform PV hosting capacity analyses in real utility feeders with and without smart inverter controls, battery storage, and EV charging. Smart inverter controls include conventional Volt/Var/Watt controls for reactive power support and active power curtailment, but also predictive controls that optimize the charging and discharging profile of the battery connected on the DC side in order to minimize the customer's economic benefit. Finally, an important result of this project is delivering an experimental setup that consists of residential-scale PV inverters with battery storage, a real-time grid simulator with an ideal voltage source as grid emulator, and micro Phasor Measurement Units (PMUs). All these components and additional software modules are coupled to one another using the FMI standard and can be co-simulated with the CyDER platform.

2. Background

The complexity of distribution grids is drastically increasing as a result of integrating larger shares of Distributed Energy Resources (DERs) including variable renewable generation, controllable demand, and storage devices with their associated controllers [1]. Solar photovoltaic (PV) systems are currently the renewable energy technology with the highest growth rate in many US states and countries worldwide. Since residential PV systems are connected throughout the distribution feeder and the peak PV generation does not necessarily coincide with the peak load demand, the power flow patterns are changing and some distribution feeders are facing voltage stability issues. Besides PV, increasing adoption of Electric Vehicles (EVs) and their uncontrolled charging can also overload existing distribution system equipment by amplifying the peak load demand in the evening hours.

The widespread adoption of PVs and EVs creates new operational and planning concerns for many Distribution System Operators (DSOs). Indicative of the general concern is the fact that the California Energy Commission (CEC) has ordered the State's investor-owned utilities to conduct Integration Capacity Analysis (ICA) on their feeders to assess maximum hosting capacities for DERs. In fact, the challenges in distribution system operation and planning with high DER penetration are multifold. One main challenge is the large variety of technologies being integrated, which imposes very different requirements on modeling and simulation tools, as well as the associated uncertainties due to variable weather patterns and user behavior. In addition, DER integration increases the coupling between distribution and transmission grids, e.g., reverse power flow, which is a challenge because the software tools used by DSOs are typically different from those used by Transmission System Operators (TSOs). Furthermore, DER integration increases the interactions and dependencies between the power system and other infrastructure systems, such as communication systems, heating and cooling systems, buildings, etc.

To address these power system challenges, a multi-domain and multi-purpose simulation tool with modular, extensible, and scalable architecture is needed. One way to obtain such a tool would be to extend existing simulation software to include all additional desired functionalities that are currently not supported, which is of course not practical. A better approach would be to develop a platform that enables co-simulation of independent models and simulators. The advantage of this approach is that specialized software and third-party tools can be leveraged to study complex interdependencies between systems while preserving simplicity, transparency, flexibility, and scalability of the simulation environment [2]. So far, several co-simulation platforms with application to power systems exist in the literature, but most of them involve only two simulation tools and are set up in an ad-hoc way without clear semantics of how tools are synchronized [3].

Recently, few co-simulation platforms that support multiple power system simulation tools have been developed. These include the Framework for Network Co-Simulation (FNCS) [4], which wraps simulation engines as agents that communicate through an FNCS broker; the platform SCEPTRE which wraps simulation tools and control devices in virtual machines that communicate through network [5]; the Integrated Energy System Modelling Framework (IEMS) that uses the discrete event system specification (DEVS) formalism [6] to wrap different simulation tools, which then communicate through a DEVS coordinator [7]; the Integrated Grid Modelling System

(IGMS) which is based on ZeroMQ [8] and defines peer-to-peer interfaces for multi-domain co-simulation [9]; the Mosaik framework which defines an Application Programming Interface (API) to be implemented by a simulation tool so it can be executed as an individual process that is coordinated by the Mosaik coordinator [10]; and the Hierarchical Engine for Large-scale Infrastructure Co-Simulation (HELICS) which defines an API to be implemented by a simulation tool so it can be co-simulated with other tools within HELICS. Other co-simulation efforts in the area of electric power systems include the work done by [11-13]. One key limitation of the existing trend in co-simulation platforms for power systems is that the development is not built on open standards. Instead, such platforms usually require simulation tools to implement custom APIs, which in turn requires ad-hoc solutions for interoperability. Furthermore, they are prone to non-determinism because of lack of rigor in the semantics, and can thus quickly become hard to maintain or obsolete because of lack of industry support.

In contrast to existing work, this project aims at developing an open-source co-simulation platform for power systems called CyDER (Cyber physical co-simulation platform for Distributed Energy Resources in smart grids), which is based on the well-defined open industry Functional Mock-up Interface (FMI) standard [14]. Even though FMI originated from the automotive industry, the goal of this project is to make it relevant for the power systems industry by coupling simulation tools in different time domains and voltage levels, such as Quasi-Static Time-Series (QSTS) distribution and transmission system simulations and real-time digital simulations involving hardware devices. Moreover, the platform enables integrating diverse tools, such as PV modules with associated inverters, EVs, and building simulators, as well as data streams coming for example from Phasor Measurement Units (PMUs). Note that the contribution of CyDER project is not in the API itself (FMI standard), but in its application to power systems co-simulations and the development of Python modules to facilitate connecting simulators and launching co-simulations with the PyFMI master algorithm [15].

3. Project Objectives

The CyDER project aims at delivering a modular and scalable co-simulation tool for power system planning and operation that will work seamlessly with existing interconnection planning tools in the utilities and be interoperable with future utility software, data streams, and controls. CyDER will maintain and enhance the efficiency and reliability of the power system in a cost-effective and safe manner, being built on three pillars: QSTS co-simulation and optimization, real-time data acquisition, and Hardware-In-the-Loop (HIL) applications. Specifically, CyDER combines transmission and distribution system simulation, data collection and analysis, power generation and load forecasting, load flexibility - primarily EV charging - and real-time control of solar PV inverters to accommodate high levels of PV penetration. Based on the FMI standard, CyDER utilizes a combination of open source tools, and commercially available platforms.

The main innovation and contribution of the project lies in adapting the FMI standard for the needs of power systems simulations by: (i) developing a library of open-source, FMI-compatible transmission, distribution and DER models - called Functional Mockup Units (FMUs) - that are relevant for the power grid industry; (ii) delivering software utilities that enable exporting new power systems simulation tools as FMUs, as well as

facilitate setting up and running co-simulations by users; and (iii) allowing co-simulation of software components (e.g. PV and EV models) with hardware components (e.g. real-time grid simulators and inverters).

Due to the focus on an industrial standard, there is high confidence that CyDER will be well received by industrial partners and get traction after project completion. One utility use case for the CyDER platform is running impact analysis of various PV inverter and storage control scenarios on the distribution grid, which in turn will contribute to creating roadmaps to achieve very high PV penetration targets (e.g. 100% in terms of peak load demand or higher). In fact, such types of analysis were central in the project and have already lead to a technical impact analysis report for PG&E.

The project is organized in three Budget Periods (BPs) and includes the following tasks, milestones, and go/no-go decision points.

- BP1
  - Tasks
    - T1.1: Development and integration of individual modules for CyDER.
    - T1.2: Predictive Analytics Module for PV, EV.
  - Milestones
    - M1.1.1: QSTS simulation success within integrated modules for PV, grid, and EV with the interfaced modules between the distribution, transmission, and FMI tools.
    - M1.2.1: Relative Root-Mean-Square Errors (RMSEs) below 30% achieved for PV forecasting and EV forecasting.
  - Go/no-go decision point: Communication of analysis data for power flow is established in QSTS, between GridDyn and CYMDIST through the FMI interface, to 5-second time step QSTS capability. Project has met milestones 1.1 and 1.2.
- BP2
  - Tasks
    - T2.1: Establishing interoperability in between CyDER modules and development of HIL setup through OPAL-RT RTDS.
  - Milestones
    - M2.1.1: CyDER has successfully coupled grid simulation, sensor data streams and smart PV inverter control to enable deterministic co-simulation.
  - Go/no-go decision point: 4-hour ahead simulation of PV, EV and grid completed for substation area, simulated to within 5% of expected generation output and synchronized to 5-second time steps. Milestone 2.1 and 2.2 are met.
- BP3
  - Tasks
    - T3.1: Multi-scenario simulation
    - T3.2: HIL testing
    - T3.3: CyDER as co-simulation platform
  - Milestones
    - M3.1.1: CyDER has successfully investigated the contribution of smart inverter controls and battery storage to increasing PV

penetration levels to >100%, on a selected substation within utility partner's territory.

- M.3.2.1: Power system FMUs communicate efficiently with the hardware setup through the Opal-RT real time simulator. Discrepancies are within 5% despite latencies and control delays.
- M.3.3.1: CyDER availability, initial maintenance provided, 5 new FMUs are created, API and accompanying documentation.

4. Project Results and Discussion

In this section, we provide an overview of realized results and compare them with the anticipated project outcomes following the structure of the Statement of Project Objectives (SOPO) and task and subtask breakdown per year.

Task 1.1: Development and Integration of Individual Modules for CyDER

Subtask 1.1.1: Design of CyDER specifications & FMI interfaces between the modules

This subtask focuses on defining the specification of the necessary inputs and outputs each module should provide to enable a co-simulation of these modules. This includes developing the FMI interfaces for the distribution planning tool - from CYME - (CYMDIST) and transmission tool (GridDYN) - developed by LLNL - as well as the PV model and inverter controls.

Distribution grid FMU (CYMDIST): We selected CYMDIST as it is commonly used by US utilities and provides a Python API, which allows it to be exported as an FMU. For this purpose, a Python function which interfaces with the Python API of CYMDIST was developed. This function allows us to start CYMDIST, and to set values to and get values from a running CYMDIST model at discrete times. This Python function was then exported as an FMU using the developed Python utility SimulatorToFMU.

SimulatorToFMU (Python utility): The approach of SimulatorToFMU [16] is to wrap the high-level Python API of a third-party simulator in a Python function, which can then be exported as an FMU. The utility uses a Python templating engine called jinja2 to generate a Modelica block that contains a wrapper which communicates with the simulation tool through its Python API. It then invokes a Modelica translator which compiles the model to C-code and exports it as an FMU. Leveraging Modelica compilers to export Python functions as FMUs ensures the backward and forward compatibility of SimulatorToFMU with older and newer versions of the FMI specification as they become available in Modelica tools. SimulatorToFMU gets as input the path to an XML file which specifies the input and the output of the simulator as well as the Python function which interacts with the simulator. It then generates a Simulator FMU which is compliant with the FMI standard.

Transmission grid FMU (GridDyn): We selected the open source simulator GridDyn as the transmission network simulator. We developed a utility to export a GridDyn model as an FMU for co-simulation 2.0. The export utility is a small executable that gets as input an XML input file (which specifies the inputs and outputs that should be

exposed through the FMI interface) along with the network model and exports the model as an FMU.

PV FMU: To model PV generation, an empirical PV model was developed in Modelica and exported as an FMU. The PV generation is computed as

$$P_{PV} = A \cdot eta \cdot G \cdot (1 - S),$$

where A is the area of the PV in m^2, eta is the lumped PV conversion efficiency including PV module and inverter, G is the incident solar irradiation at the tilted surface, and S is the percentage of shaded area. To compute the incident solar irradiation, the model uses historical weather data in the form of a Typical Meteorological Year (TMY) weather data file. The weather data is fed to a detailed sky model along with tilt, azimuth, latitude, longitude, and time zone to compute the incident solar irradiation on the PV surface.

Inverter control FMU: Smart inverter controllers, namely Volt/Var and Volt/Var/Watt controls, were modeled in Modelica and exported as FMUs. A detailed description of the controls will be provided later in this report (Task 3.1), together with the multi-scenario simulation results.

EV FMU: To model the effect of EVs on the grid, we created an FMU that simulates the power demand of an EV fleet based on individual vehicle information such as parking duration (hours), maximum charging power (kW), and energy needs (kWh).

Subtask 1.1.2: Data Collection and interface for Grid, PV

The goal of this subtask is to collect data for various sources to integrate in CyDER. Our partner SolarCity provided 10,000 PV inverter production profiles (with 15-minute resolution) for 2015, as well as associated zip code information for 7,000 of these profiles. In addition, our partner ChargePoint provided 3 month of charging session data from EVs around the San Francisco bay area. The data contains the following information for each charging session: ID, device ID, Name, Driver ID, City, County, State, zip code, session start time, session end time, category, max. # kWh, and session time (minutes).

In addition, our partner PG&E provided detailed models for 52 real feeders connected to 8 different substations in their service area. Besides the topology files, the data include SCADA active and reactive power measurements at the feeder head for 2016, as well as long-term scenarios for load demand and installed PV power. Fig. 1 shows the topology of a substation with each feeder represented by a different color (left plot), as well as the voltage profile result of a power flow simulation (right plot). Besides data from partners, the LBNL team integrated data streams from LBNL's FLEXGRID test facility. Specifically, available data include measurements from a micro-PMU and three single phase PV inverters. Fig. 2 gives a high-level overview of the CyDER platform including integrated models, simulators, and data streams.

Subtask 1.1.3: Development of QSTS Solution, and visualization platform

Selecting and setting up the master algorithm for CyDER to support QSTS simulation is in the core of this task. Different options for the master algorithm were considered including: (i) Ptolemy II [17], (ii) GridDyn itself for T&D co-simulation, (iii) OMSimulator in programming language Lua, and (iv) PyFMI [15]. Due to PyFMI's advantages related to user friendliness and the fact that it is backed up by a company (Modelon), which provides confidence of continuous upgrade and support, the project team selected PyFMI as the master algorithm for CyDER to link, exchange, and

synchronize data between the FMUs at runtime. PyFMI is an open-source master algorithm written in Python. It provides functions which enable interfacing with FMUs. The main functions are unzipping, loading, instantiating, initializing, and running the FMUs, while synchronizing the data exchange between the FMUs. The code snippet in Fig. 3 shows how PyFMI is used to link two FMUs. The steps involved are:

- Load the FMUs (Line 9 and 10)
- Create an object which contains the FMUs instances (Line 12)
- Define the connection between the FMUs (Line 15)
- Create an object which represents the coupled system (Line 18)
- Simulate the coupled system (Line 21)
- Retrieve the simulation results (Line 24 and 25)

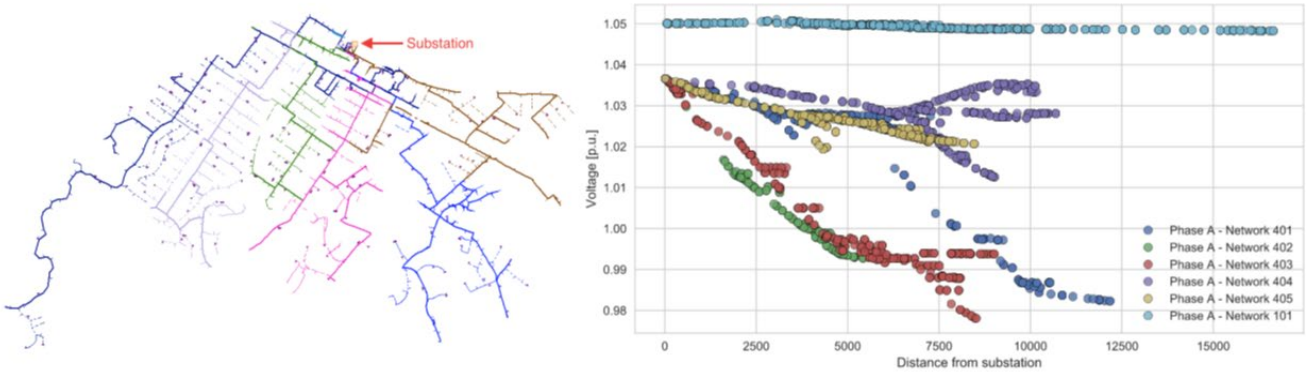

Figure 1: Topology of a PG&E substation model with six feeders (left) and voltage profile (right).

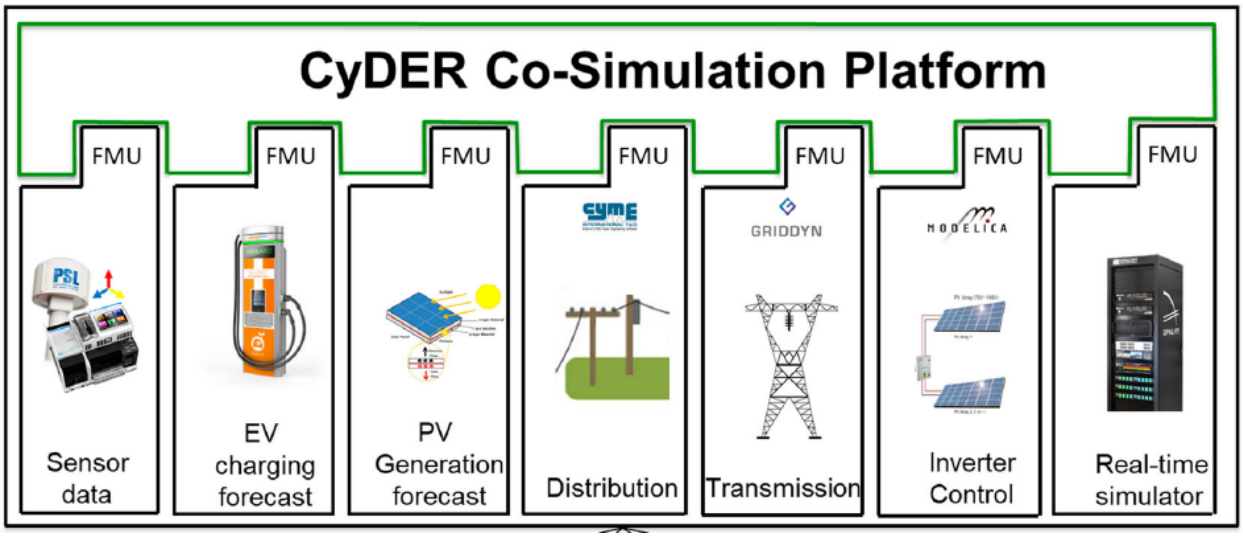

Figure 2: Overview of CyDER platform connecting various third-party models, simulators and data streams that support the FMI standard and are exported as FMUs.

```
from pyfmi import load_fmu
from pyfmi.fmi_coupled import CoupledFMUModelME2

# Simulation parameters
start_time = 0.0
stop_time  = 0.1

# Loading the FMUs
cymdist=load_fmu("cymdist.fmu")
control=load_fmu("control.fmu")

models = [("cymdist", cymdist), ("control", control)]

# Creating connection list
connections = [(cymdist, "v", control, "v"), (control, "Q", cymdist, "Q")]

# Creating the coupled system
coupled_simulation = CoupledFMUModelME2 (models, connections)

# Running the simulation
res=coupled_simulation.simulate(start_time=start_time, final_time=stop_time)

# Retrieving the trajectories
sim_time=res["time"]
cymdist_v=res["cymdist.v"]
```

Figure 3: Example of setting up and running a co-simulation in CyDER using PyFMI as master.

PyFMI supports the two FMI simulation models, namely the Co-Simulation (CS) mode and Model Exchange (ME) mode. In the ME mode, the FMU exposes the derivative of the state as output and thus an external integration solver is needed to perform numerical integration, which typically comes with the master algorithm. In contrast, in the CS mode, the FMU contains an integrator and already exposes the integrated state as output. In general, the ME mode is more versatile than the CS mode and more efficient in capturing events using a variable time step. The section on Task 3.3 will elaborate more on the two modes and the experiences from this project.

A main issue in co-simulation is handling algebraic loops that appear when connecting multiple simulators and models. In this project, there were mostly two types of algebraic loops that needed to be resolved. The first one is an algebraic loop between the transmission grid model and the distribution feeder models. In general, this algebraic loop occurs because the aggregate power demand of distribution loads depends on the voltage at the feeder head, which in turn depends on the aggregate power demand. In this project, this algebraic loop was broken by adopting a ZIP-load type representation of the distribution grid when simulating the transmission system. Specifically, the parameters of the ZIP load are estimated by simulating the distribution grid with three different voltage head values [18]. The second type of algebraic loop is the one occurring between the distribution grid and the inverter Volt/Var/Watt controllers. The reason is that the control activation depends on the voltage at the inverter's point of common coupling, which in turn depends on the controlled active and/or reactive power. Since this is a more dynamic interaction, the strategy to break this algebraic loop was to introduce a first-order transfer function in the inverter active and reactive power output, which represents a delay in reaching the desired power setpoint.

Even though the development of a web-based visualization platform to report results from the QSTS co-simulations was included in the initial scope of work, this part was removed during the course of the project. Instead, a simpler visualization functionality was developed based on a Jupyter notebook and excel files, as explained in Task 3.1.

Task 1.2: Predictive Analytics Module for PV, EV

Subtask 1.2.1: PV generation forecasting

Two different approaches were developed in the paper for PV power forecasting. The first one was developed in the first year of the project with the goal of providing input to the stochastic QSTS simulations of Subtask 2.1.1. It uses a NARX (Nonlinear AutoRegressive with eXogenous inputs) Artificial Neural Network (ANN) model to predict values from multiple inverters. The NARX model enables taking into consideration historical weather data coupled with historical PV patterns. The Fast Artificial Neural Network (FANN) library in C was used to implement this model for CyDER. The solar dataset mentioned in Subtask 1.1.2 was used for training and validation purposes and it contains 350,400,000 data points in total across all inverters. This forecast model was able to deliver 4-hour ahead forecasts with a normalized RMSE smaller than 15% and 24-hour ahead forecasts with normalized RMSE less than 30%.

The second approach was developed during the last year of the project (but is included in this section for coherence) and the main application is Model Predictive Control (MPC). Specifically, we designed a hybrid forecasting method for short-term PV power forecasting (prediction horizon from minutes to 1 day) that combines a Seasonal Auto-Regressive Integrated Moving Average (SARIMA) model with an ANN model using weighing factors computed periodically via a least squares problem. The overall structure of the hybrid forecasting model is shown in Fig. 4. This modular approach offers practical advantages for use in MPC applications compared with previous work. Specifically, SARIMA and ANN forecasts are independent to one another, which increases reliability in case one of them fails, e.g., due to gaps in measurements. Furthermore, our approach optimally combines individual forecasts within a prediction horizon of 24 hours, which is typical for MPC, and enables re-computing the weighting factors periodically to improve performance. Finally, the approach can easily integrate additional prediction models, which only requires re-computing of the weighting factors.

The prediction models were trained with data from one of the three single-phase PV systems of FLEXGRID, and then used to forecast the PV power production. Fig. 5 shows the day-ahead forecasts obtained at 00:00 of each day by SARIMA (in blue), ANN (in magenta), the hybrid model (in green), and actual values (in black). The data span the period from September 30 to October 7, 2018. In addition, the 1-step ahead forecasts with the hybrid model are shown with the red, dashed curve. According to the results, the proposed hybrid method reduces forecast error by 10% compared with the individual models during periods with increased volatility in PV power. The final normalized RMSE was found to be in the range 5-10% for 24-hour ahead predictions.

Besides PV forecasting, the hybrid method of Fig. 4 has also been used for short-term load forecasting. While the overall structure of the model is the same, the architecture of

the ANN was changed to include the day-of-week and hour-of-day dependencies in load power, which was shown to reduce RMSE by around 7%.

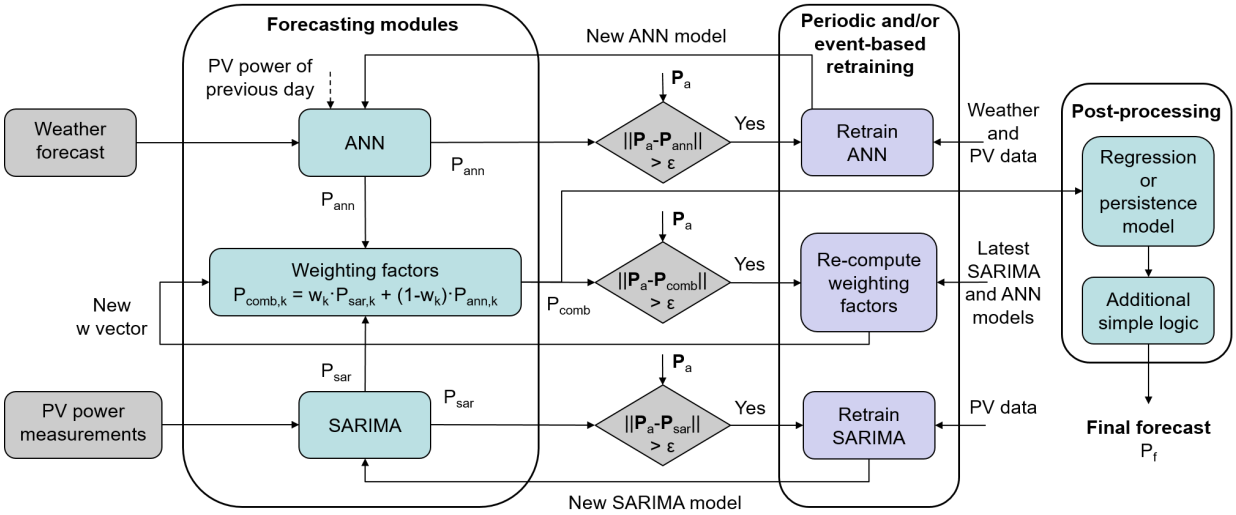

Figure 4: Schematic of hybrid approach for PV power forecasting including the individual models (SARIMA, ANN), the weighting factors, and a post-processing unit [19].

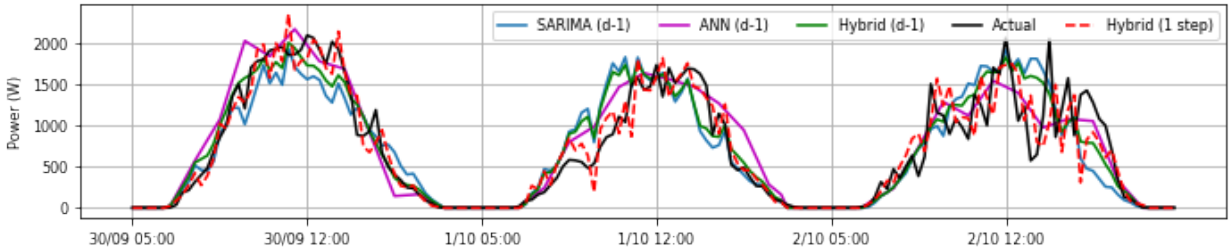

Figure 5: Actual values and PV power forecasts by SARIMA, ANN and hybrid models.

Subtask 1.2.2: EV forecasting

The goal of this task was to forecast the aggregate EV power demand using occupancy data (number of vehicles parked) as an input to a V2G-Sim simulation. This work was motivated by the need to provide input to the stochastic QSTS simulations of Subtask 2.1.1. V2G-Sim is a Python tool that provides bottom-up modeling from individual vehicle dynamics all the way up to aggregate grid impacts and opportunities. Nevertheless, to use V2G-Sim various details about the vehicle and charging station composition is needed (e.g., vehicle mix and powertrain specifications, charging station mix and power level specifications, etc.), which were not available in the dataset (as mentioned in Subtask 1.1.2, the available data include charging power and EV occupancy data). For this purpose, we used 70% of the available data as training set and ran iterative V2G-Sim simulations to estimate such unknown parameters. Afterwards, unobserved EV occupancy data (the remaining 30%) were provided as input to the trained V2G-Sim simulation to forecast the total EV power demand. Indicative results are shown in Fig. 6 and the normalized RMSE was found to be less than 10% for all months.

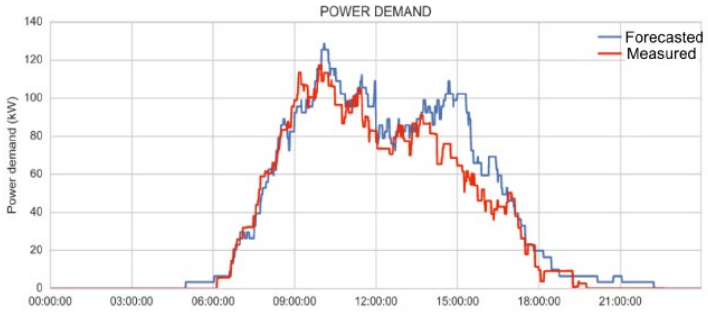

Figure 6: Measured and forecasted EV power consumption at a feeder in San Jose.

Overview of project status at the end of BP1: The relevant milestones M1.1.1 and M1.2.1 were successfully met. The project passed the Go/no-go decision point, as M1.1.1 and M1.2.1 were met, developed FMUs passed the FMI compliance check using the FMU checker tool, and the communication among the various FMUs down to a 5-second time step was possible. In contrast to the initial plan in the SOPO, PyFMI was used as the master algorithm instead of Ptolemy II due to the advantages mentioned previously.

Task 2.1: Establishing interoperability in between CyDER modules and development of hardware in the loop (HIL) setup.

Subtask 2.1.1: Real-time stochastic time-series solution and Hardware-in-the-loop module interface

This task is primarily concerned with connecting all CyDER FMUs in order to be able to run co-simulations including simulated PVs, EVs, distribution grid model, and transmission grid model. For this purpose, the distribution grid FMU and the SCADA data are combined to come up with time series of active and reactive power for each load node of the feeder. Specifically, we applied the methodology used by PG&E according to which the total active and reactive power at the feeder head (SCADA data) is allocated to the downstream loads proportionally to their relative size. The distribution FMU was then updated to reflect this new workflow. At every step, the distribution FMU:

- Sets the reference voltage at the feeder head (input from the transmission FMU)
- Runs a load allocation to change the power demand of each load using the closest SCADA data point from the current simulation time.
- Adds new loads or PVs at specified locations and sets their power demand using input values sent to the FMU.
- Runs the power flow.
- Gets the new state of the network including current magnitudes and phases at the feeder head (input to the transmission FMU).

A Volt/Var control FMU and a Volt/Watt FMU were developed in Modelica for use in the simulated and real PV systems, respectively. Moreover, the EV FMU was modified to provide the opportunity to apply a Volt/Watt control to reduce voltage issues on the grid by shifting power demand from EVs, while preserving energy constraints set by EV owners. The left plot of Fig. 7 shows different paths to charge an EV during the time it is parked (it can charge as soon as possible, as late as possible, or with a constant power demand), and the right plot shows the effect on total EV power demand.

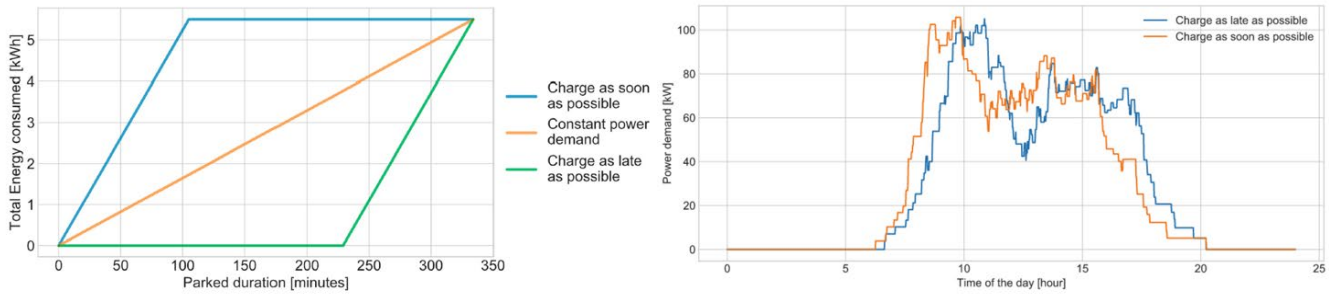

Figure 7: Flexibility in EV charging profile (left) and effect on aggregate demand (right).

The various FMUs are connected as shown in Fig. 8. Note that an IEEE 13-bus system was used to represent the transmission grid. The transmission FMU outputs the voltage at the substation (feeder head), which is then used as an input to the distribution FMUs that in turn return the current injection. The substation includes 6 feeders with a total peak demand of 10 MW and there are three PV systems modeled each with an installed power of 350 kW, as well as three EV charging stations connected at the PV modes, each with 100 kW peak power demand. Simulations were performed for three cases: (i) uncontrolled PV, (ii) PV with Volt/Var control, and (iii) uncontrolled PV and controlled EV charging. Simulation results for these three cases are shown in Fig. 9. As expected, cases (ii) and (iii) reduces the voltages compared with case (i). Observe that the voltage is reduced also at night time for case (ii), because the inverters can provide reactive power compensation even without any active power production.

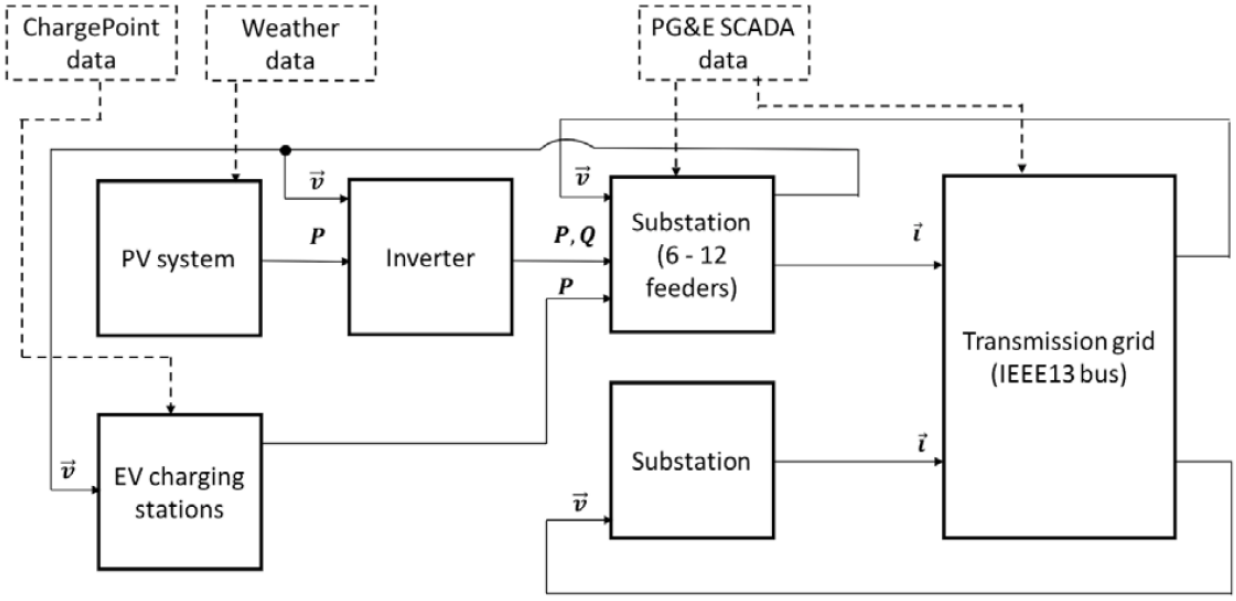

Figure 8: Overview of the connection of various FMUs in the co-simulation.

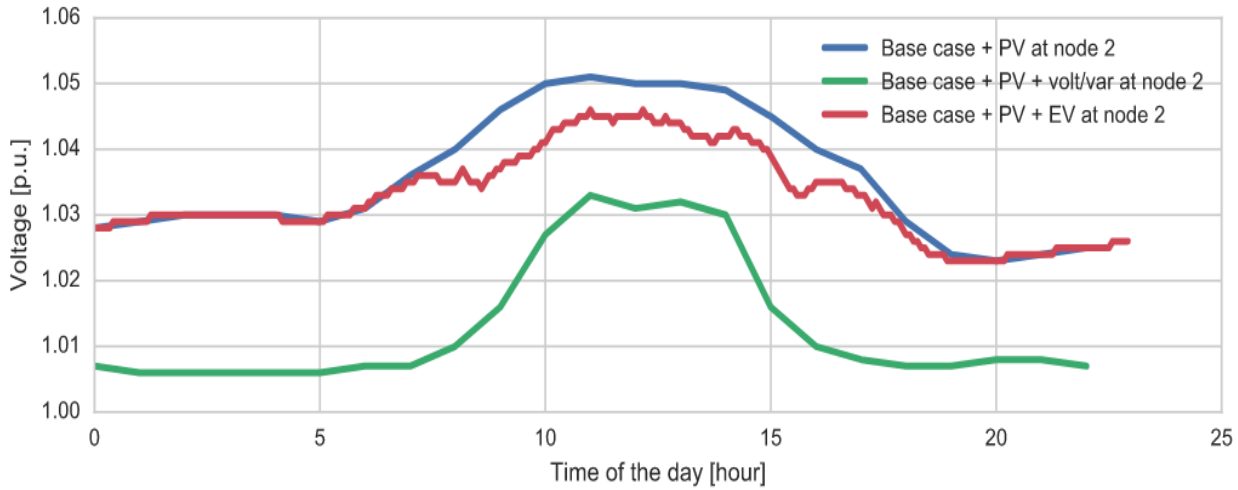

Figure 9: Voltage profiles for the three simulated cases.

Moreover, preliminary multi-scenario simulations were performed in this task (note that this type of analysis is the primary focus of Task 3.1). Specifically, we took one of the PG&E feeder models and simulated the following scenarios:

- Use case 1: Uncontrolled PVs
- Use case 2: PVs with Volt/Var control
- Use case 3: Uncontrolled PV and uncontrolled EV demand based on a simulated profile using National Household Travel Survey (NHTS) data
- Use case 4: Uncontrolled PV and uncontrolled EV demand based on ChargePoint measurements
- Use case 5: PVs with Volt/Var control and uncontrolled EVs (ChargePoint data).

Simulations covered multiple PV penetration scenarios in the range [0,100]%. Note that PV penetration of 100% means that the annually produced PV energy is equal to the annual load energy demand.

Fig. 10 presents the highest voltage in the grid as a function of the PV penetration level. The solid line represents the median for the first 2 weeks of June, while the shaded area represents the 80% percentile. The co-simulation shows that in absence of controls, a 20% PV penetration leads to over-voltages (voltage above 105% of the reference value). In use case 2, smart inverters consume reactive power equal to 30% of the PV VA-rating at peak production, if the voltage is above 107% (PG&E Volt/VAR control), thus reducing over-voltage from 1.19 p.u. to 1.13 p.u. at 100% PV penetration. However, note that the increase in PV hosting capacity - while keeping the worst-case voltage below 105% of the reference - with reactive power compensation only is limited (around 2%). In use cases 3 and 4, even though the EV load demand is uncontrolled, it still absorbs some of the PV active power, enabling PV penetration to 30% and 40%, respectively, without leading to overvoltages. A larger PV penetration without voltage problems can be obtained in use case 4 because the ChargePoing EV demand is shifted at noon and in the afternoon in order to coincide with the peak PV generation. In use case 5, EV demand and reactive power absorption reduce the over-voltage from 1.19 p.u to 1.10 p.u at 100% of PV penetration and increase the PV hosting capacity to more than 40%.

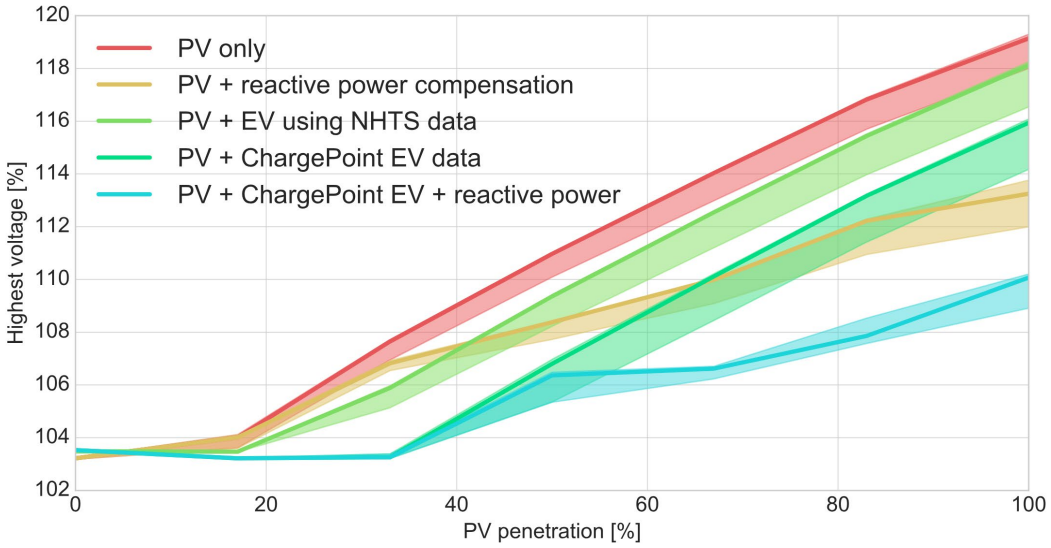

Figure 10: Result of the co-simulation where over-voltages (voltage levels above 105% of the reference) are represented as a function of the PV penetration.

Within this task, the team also used the tool PSUADE (Problem Solving Environment for Uncertainty Analysis and Design Exploration) developed by LLNL to automate the execution of CyDER co-simulations and produce output for analysis. Specifically, PSUADE executes multiple CyDER QSTS simulations with varied specified input parameters (for this purpose, CyDER was exported as a stand-alone tool to be used by LLNL, and will be called “CyDER black box” in the following). Four scenarios were generated by varying solar PV and load scenarios for a time window of 4-24 hours in the future. More precisely, three 4-h-ahead and one 24-h-ahead PV forecast scenarios under 5 different levels of PV penetration were studied.

The CyDER black box has four input and four output parameters. The input parameters are (1) feeder IDs; (2) node IDs for PV locations; (3) the added PV generation per node for a given % PV penetration; and (4) the date of the studied day. Additionally, the black box produces several hourly output parameters, which are (1) voltage at the nodes with PVs; (2) total PV generation at the PV sites; (3) load and net load on feeders; and (4) the worst-case high and low voltage per phase from across all the nodes. In stochastic QSTS simulations over a day, the solar PV data, which is captured at 15-minute time intervals using historical data and insolation forecasts, is used. In addition, time-varying load data that is coincident with time-varying PV output data is utilized. Five PG&E feeders were identified and studied in stochastic QSTS simulations. On each feeder, 10 locations (i.e., nodes) were randomly selected for PV installation. For simplicity, the installed PV generation across the selected nodes on each feeder is uniformly distributed for a given % PV penetration. Also, the total installed PV generation on each feeder was calculated based on the daily peak load active power on each feeder.

The plots in the upper panel of Fig. 11 show the worst-case high and low voltage values (in p.u.) and feeder loads and PV generation (in kW) for a 4-h-ahead PV forecast scenario. The worst-case high and low voltage values from across the installed PV locations are shown in the upper-right panel. One of the key observations is that the lowest voltage values are pretty much the same as we increase the PV penetration from 15% to 100%. Meanwhile, the highest voltage values start increasing after 50%

penetration of PV. The top of the lower panel shows the daily voltage profiles for the nodes where PVs are installed for different levels of PV penetration, with the lower and upper bounds for the voltage values as well as their median. In the bottom of lower panel, the daily feeder load and net load profile are plotted with respect to % PV penetration. At 75% and 100% penetration of PV, we observe a negative net load around noon and early afternoon, which is indicative of reverse power flow. Note that the team investigated the option of exporting PSUADE as an FMU in order to integrate it within CyDER. However, due to the change in scope of work after the BP2 phase review meeting, no further efforts were planned in this direction.

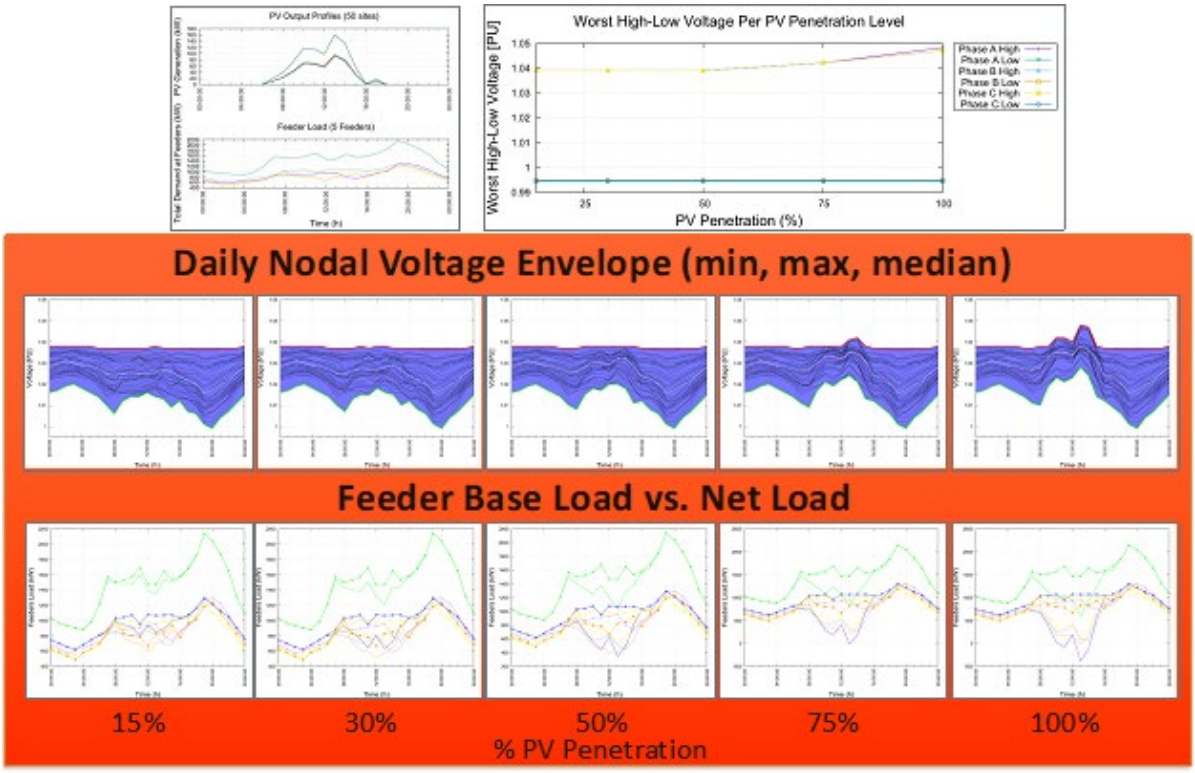

Figure 11: Stochastic QSTS simulation results with 4-h-ahead PV forecast data (PSUADE).

In addition, this task developed the initial capability to perform HIL co-simulations involving both software and hardware components. The HIL simulations were performed at LBNL's FLEXGRID test facility (Fig. 12), which includes three single-phase PV inverters, a load emulator, an OPAL-RT real-time grid simulator, an Ametek grid emulator (i.e. ideal voltage source), and a micro-PMU. In this section, we present the preliminary HIL test results obtained in BP2. More comprehensive HIL simulations were performed in the last year of the project and are reported in Task 3.2.

Before starting the HIL testing, several preparation tasks needed to be carried out including interconnection of grid simulator and emulator (OPAL-RT and Ametek), development of the real-time simulation model, finalizing development of FMUs to interface with real equipment (emulator, inverter, sensors, control). Furthermore, we selected one of the available PG&E feeder models (feeder BU0001) and converted it from the initial CYMDIST format to the ePHASORSIM format required for HIL simulations.

Finally, some preliminary HIL tests with Ametek energized were carried out to verify system's stability before running the actual CyDER tests.

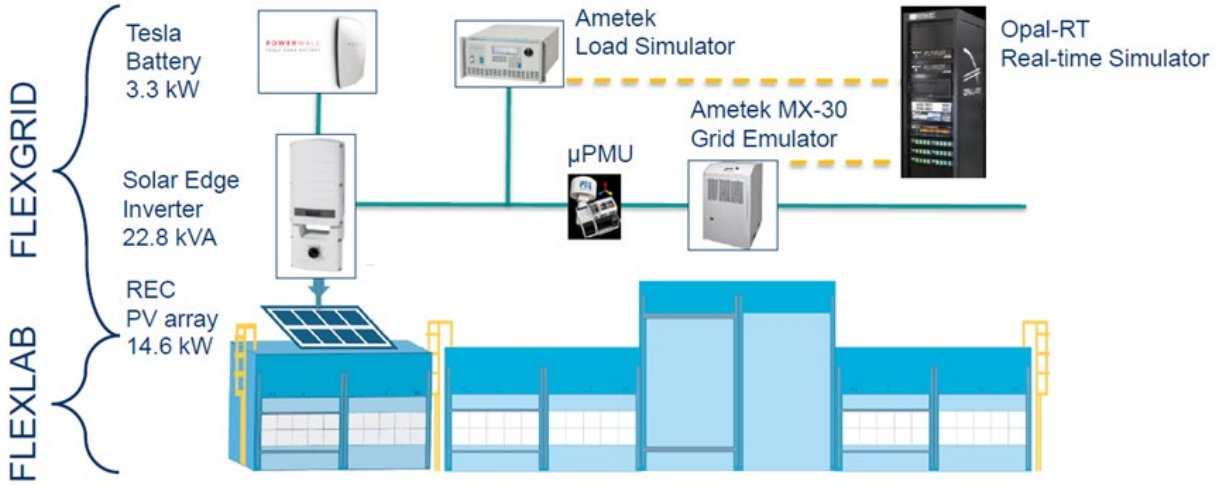

Figure 12: LBNL' FLEXGRID test facility that hosted the HIL experiments with CyDER.

Fig. 13 shows the software and hardware components included in the HIL setup. The center of the schematic is the real-time grid simulator from OPAL-RT. All components to the right are hardware devices installed at FLEXGRID, whereas the components to the left are software modules and part of CyDER. The grid simulator is encapsulated in a custom FMU, called Simulator API FMU, to allow simple utilization and configuration. The backend of the HIL simulation consists of two components, OPAL-RT, the real-time simulator, and the master PC. The real-time simulator exclusively communicates with the master PC to obtain models and adapt parameters within the model. The master PC allows Python API calls, which are utilized within the Simulator API FMU. In the first step of initialization, the provided distribution grid model is loaded onto the simulator to set up and start the HIL test. Nodal voltages are written to analog outputs to control the Ametek grid emulator. Current measurements of the resulting power flow are fed back to the model via analog inputs of the Opal-RT.

The three simulated PV inverters are controlled with an emulated Volt-Var control, running as a standalone Inverter Control FMU, whereas the actual inverters utilize the same FMU but parameterized as a Volt-Watt control. Volt-Var or Volt-Watt controls vary the reactive or active power output based on local system voltage. The communication between Inverter Control FMU and Simulator API FMU is defined by the FMI standard, and basically allows bi-directional data transfer, which are measured system parameters as inputs and inverter setpoints as outputs in this case.

Due to an internal equipment failure of the Ametek MX-30 grid emulator (malfunction of an internal resistor), the test setup in BP2 was revised to keep full testing capacity of the CyDER platform without the ability to control the system voltage. Specifically, the HIL test was conducted as open loop, i.e. without Ametek, with the PV system connected to the LBNL grid. In this setup, the real-time simulator reads the active power generation of the PV system and feeds it into the feeder model in ePhasorsim. While the initial test intended to utilize the micro-PMU readings from the uPMU API FMU,

for this revised setup the voltages must be acquired the same way as for the emulated PV systems, via the Simulator FMU.

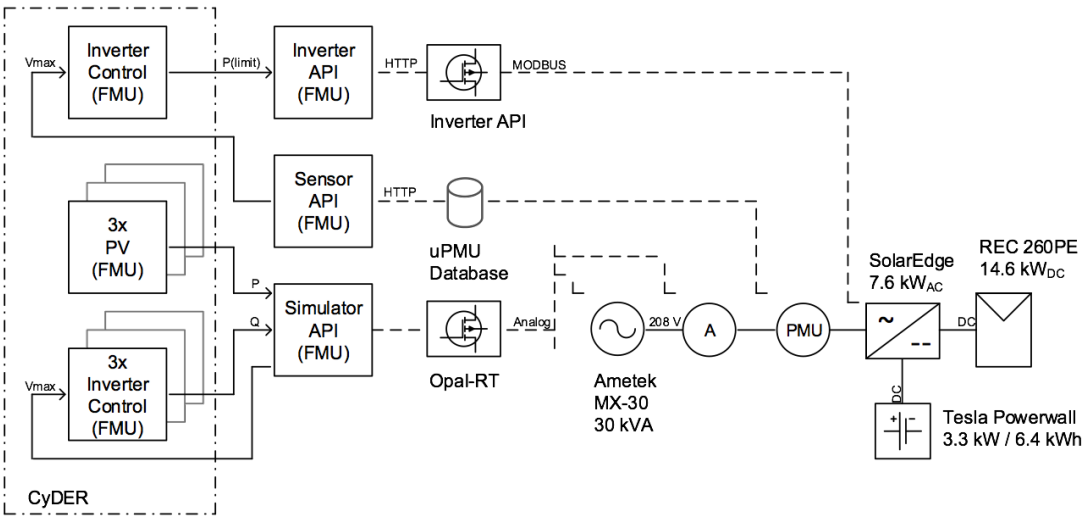

Figure 13: HIL setup for a combined software and hardware co-simulation with CyDER [21].

The three simulated PV systems use TMY weather data for San Francisco and are orientated towards West, South, and East, respectively, with a tilt angle of 10 degrees towards the orientation. The PVs are configured to a rated power of 2 MW and the HIL simulation is performed for a day in June with high solar irradiance. The real PV system is scaled from 15 kW up to 0.9 MW to create high voltages and trigger activation of controllers. The loads within the model are set based on a load allocation within the CYMDIST feeder model, where real SCADA data is used to allocate the demand, based on the power rating in CYMDIST. The allocated load profiles are then duplicated for each line, which basically triples the power flow, allowing for higher PV generation to ultimately force high voltages on the feeder. The date of the loads from SCADA data is again chosen to match the month of the test, and reflect a typical load profile.

The HIL test was executed for almost a full day (23 hours) with a CyDER time step of 5 minutes. While CyDER computes much faster in simulation mode, the addition of the real-time simulator FMU limited the time step within CyDER. However, 5 minutes is still below the design time step of the HIL test of 15 minutes. Besides the 5-minute time step for CyDER, the HIL test included a variety of sub-models which run in different time steps. These include the real-time HIL model in a 20 µs time step, the real-time grid model in a 10 ms time step, and the load change within the model in a 15-minute time step. Fig. 14 shows the results of the test as power flow and control actions taken by CyDER.

The results are shown as a set of four time-series plots. The first plot shows the feeder base load, which is allocated from SCADA data, in purple, the simulated PV generation profiles for the orientations of East, South, and West in green, orange and red, respectively, and the real PV generation at FLEXGRID, in blue. The different time steps of the system can be observed by the step function changes in the corresponding time-series. The power flow from the various sources feeds the real-time simulation where the nodal voltages can be retrieved via the Simulator FMU, as shown in the second plot. These voltages are again subject to CyDER's 5-minute time step. It can be seen that due to the scaling of the loads, the nodal voltages for the real PV and simulated PV2 are

actually below the 0.95 V p.u. threshold when no PV generation is present. However, once the sun rises, the PV plants increase the nodal voltages by about 10%, which leads to high-voltage periods.

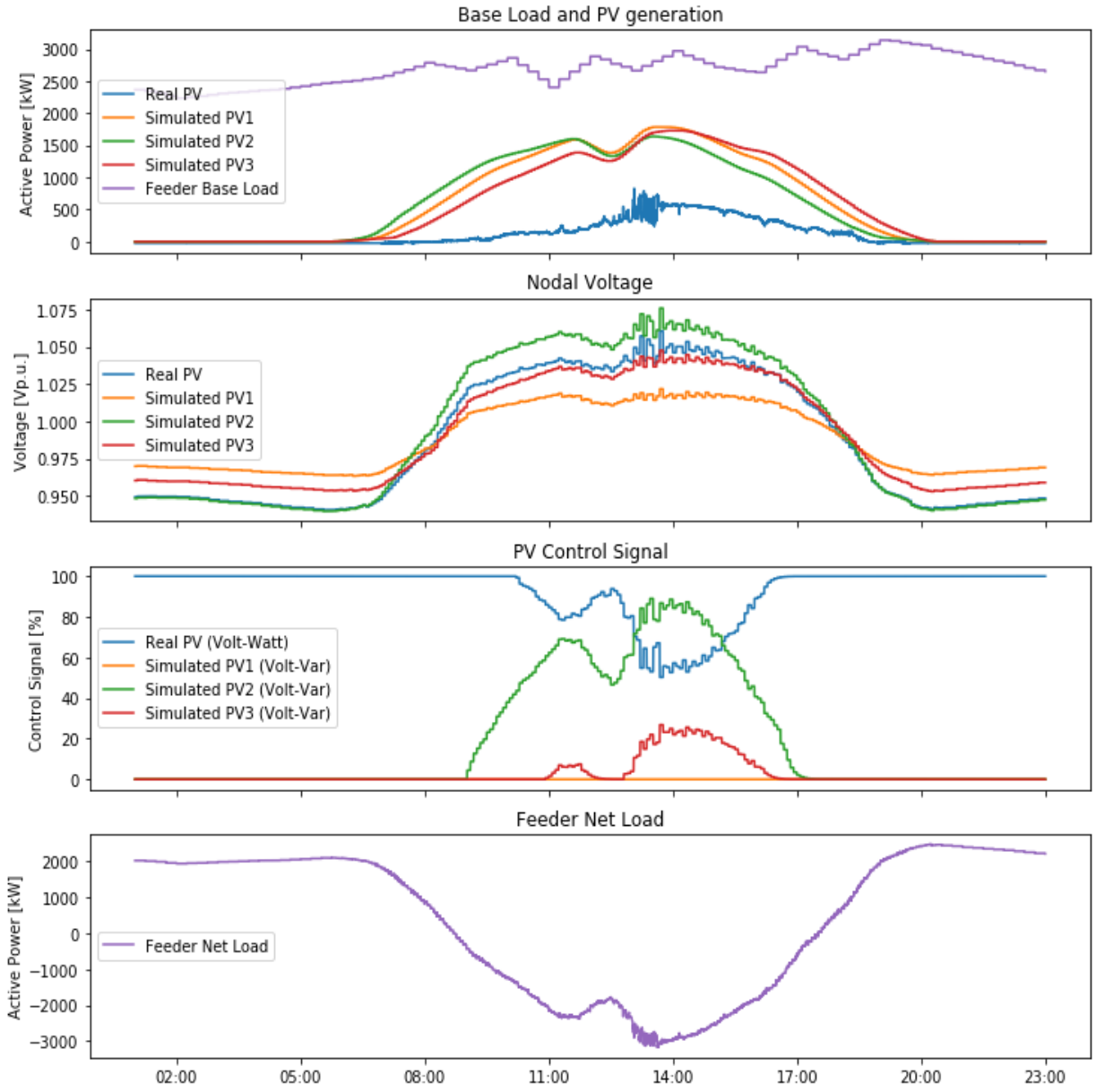

Figure 14: Time-domain HIL testing results [21].

For a period from 9 AM to 5 PM, the Volt-Var or Volt-Watt control kick in by absorbing reactive power, or curtailing active power, in case of the real PV system. The control signals can be observed in the third plot. Depending on the location of the node along the feeder, and power demand of surrounding loads, the control signals differ. While the simulated PV2 system absorbs reactive power all day long, due to the high nodal voltages, the simulated PV1 does not provide any reactive power support. The real PV system was controlled down to around 50%, which actually curtailed some generation during the peak hours. The last plot shows the feeder load at the feeder head, as an

aggregation of all loads and generation on the feeder. It can be seen that starting from 9 AM to 5 PM reverse power flow is apparent. Roughly speaking, it can be stated that the test was concluded with a 200 % PV penetration scenario.

Subtask 2.1.2: Connect Grid, Communications, sensor data, inverter, and predictive analytics for sensor placement algorithm

Overall, this task focuses on establishing connections among the various FMUs and the associated data streams. One example is connecting the PG&E feeder models with the corresponding feeder head SCADA data, as well as performing load allocation to derive load demand profiles for each of the load buses, as explained in the previous subtask. Additional work performed within this task includes linking the micro-PMU measurements to the BTrDB database, establishing an API to read measurements and write control setpoints to the PV inverters of FLEXGRID, as well as an interface to query weather data and fuse them with PV inverter data to provide input to the forecasting functions.

Furthermore, two micro-PMUs and associated communication devices were delivered to PG&E. One of the PG&E feeders presented an obvious use case for sensor placement: this particular feeder had a 4 MW / 120MWh battery located close to the edge of the feeder, but also back-tied with an edge of another feeder, participating in a frequency regulation market at the transmission level. The 4-second regulation signal and response of the battery result in fluctuation in the voltage at the feeder head. Such high-frequency variations could not be captured by the existing SCADA system. Therefore, PG&E and the team decided to place one micro-PMU at the feeder head, and another at the battery, to get a precise estimation of the feeder's net load.

Fig. 15 shows the net feeder load (true load profile with the battery operation excluded) estimated using data from the two micro-PMUs. The plot also includes the SCADA measurements for the battery and the feeder breaker to calculate the SCADA net load in purple. Due to unsynchronized measurements and interpolation, the resulting net load curve from SCADA data is confounding. Specifically, the net load curve appears to be much more volatile than it really is. In contrast, the computed net load profile based on micro-PMU data (shown in blue) is more accurate due to measurements with higher sampling rate at both locations.

Subtask 2.1.3: Commercialization

The original scope of work included some effort to commercialize the CyDER platform. However, this subtask was removed at the end of BP2.

Subtask 2.1.4: Coordination with Other SuNLaMP and GMLC Projects

LBNL hosted a workshop on co-simulation platforms on May 21 2018, which was attended by representatives from DOE, LBNL, LLNL, SLAC, SNL, NREL, PNNL, and ANL, covering multiple SuNLaMP and GMLC projects (IEMS, IGMS, FNCS, HELICS, SCEPTRE, VADER). A number of interesting experiences across different projects were shared and a result of the workshop was the decision to join forces between the CyDER and HELICS projects. This was organized as a new subtask in the third year of the CyDER project and the main objective was to integrate in HELICS FMUs developed within the

CyDER project, and verify that the HELICS co-simulation returns the same results as the one in CyDER. A summary of this effort will be provided later in this report (Task 3.3).

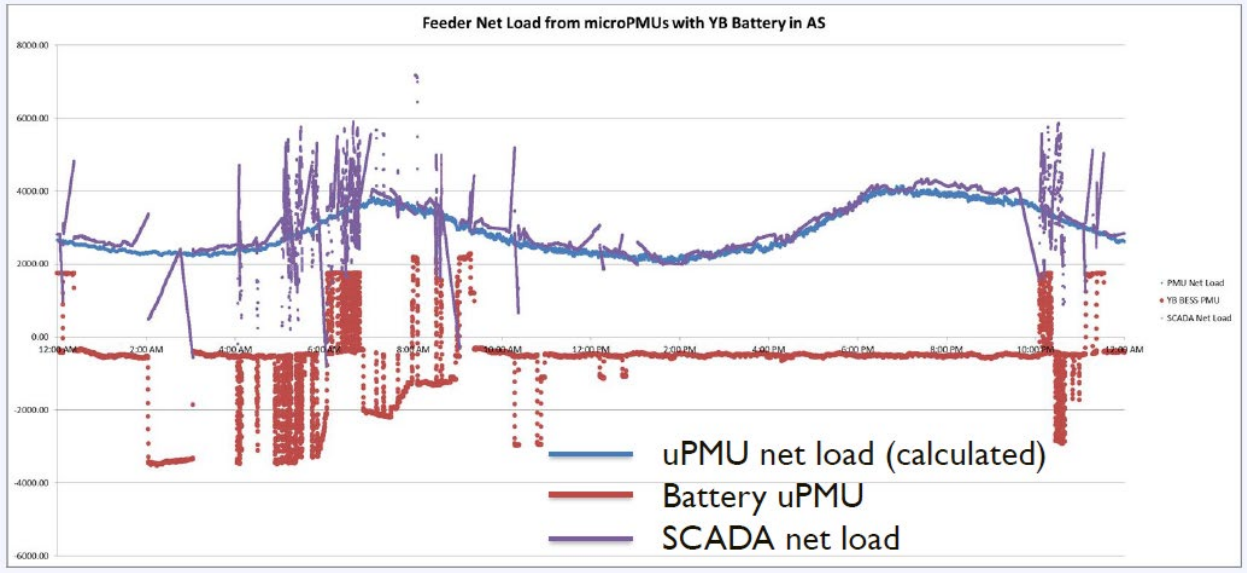

Figure 15: PG&E feeder net load calculated by SCADA measurements, as well as by measurements from two micro-PMUs, one at feeder head and the other at the battery.

Overview of project status at the end of BP2: The relevant milestone M2.1.1 was successfully met. The project passed the Go/no-go decision point, as M2.1.1 was met, 5 scenarios of PV penetrations were included in 4-hour and 24-hour ahead simulation, the first version of the HIL testing was performed, and the results from computer simulation and HIL simulation were within 5% accuracy. In contrast to the initial plan, the first HIL test was performed in an open-loop setting, i.e. without energizing the Ametek grid emulator, due to an equipment failure. As explained previously, the team was able to come up with an alternative plan to ensure successful completion of this task.

Task 3.1: Multi-scenario simulation

This task brings together results from the first two years of the project, as well as newly developed capabilities during the third year, to support various types of simulations involving large penetration of PVs and EVs in distribution grids. The rest of this section is organized in subsections that summarize individual pieces of work.

Analysis of PG&E feeder dataset

We conducted a feeder model analysis to ensure that we are able to represent a variety of scenarios. The analysis shows that we have access to a variety of customer types (Fig. 16): 50% of the feeders are dominated by residential customers, while the rest are dominated by industrial (40%), agricultural (4%) and special load customers (5%). Feeder length varies from 2 to 64 km with 68% of the feeders under 10 km.

Figure 16: Energy demand breakdown by customers for each feeder.

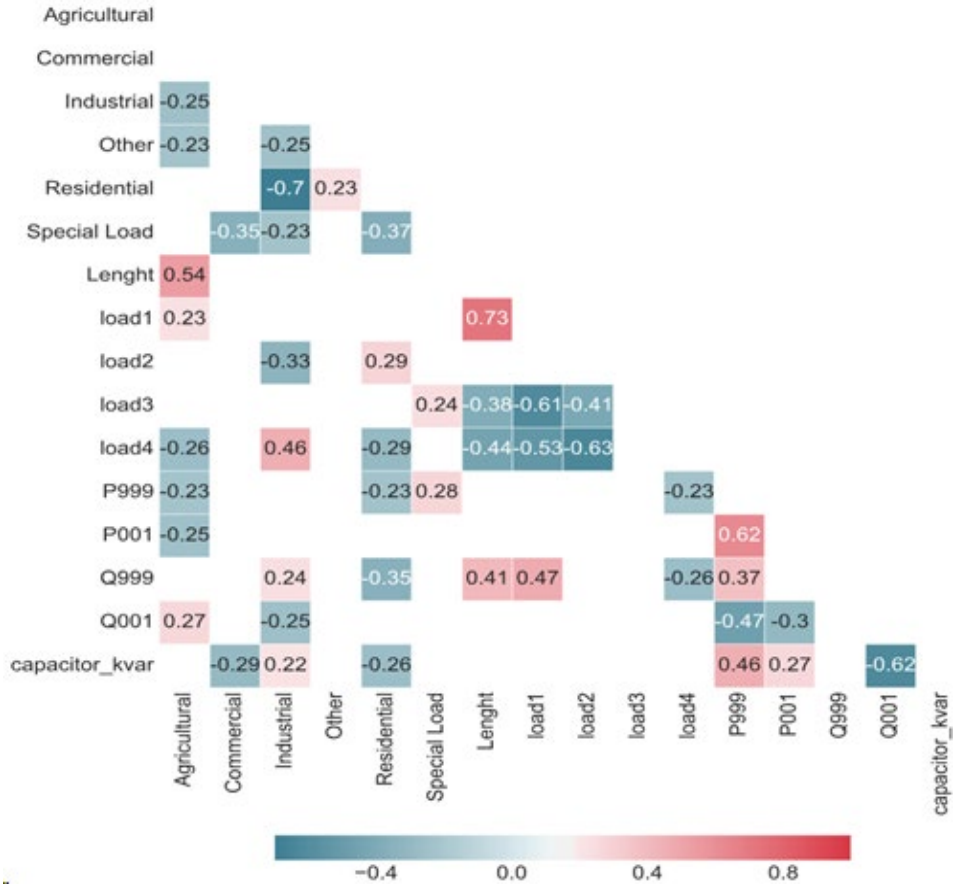

Figure 17: Correlation matrix between feeder's characteristics (correlations are only displayed for absolute values higher than 0.2).

Fig. 17 shows the correlations between feeder characteristics, for instance:

- Agricultural customers and feeder length (correlation of 0.54).
- Feeder length and percentage of energy demand in the first 25% of the feeder length (correlation of 0.73).

- Industrial customers and percentage of energy demand in the last 25% of the feeder length (correlation of 0.46).
- Residential customers and reactive power peak demand (correlation of -0.35).
- Feeder length and reactive power peak demand (correlation of 0.41).
- Capacitor banks and active power demand (correlation of 0.46).

In addition, we were able to access a public database (login required) containing information for more than 3000 PG&E feeders. The database contains data on feeder peak demands, as well as maximum capacity in MW and maximum PV and EV capacity in various scenarios. PG&E's feeders span from less than 1 MW to more than 20 MW peak demand, with a significant number of feeders around 8 MW peak demand. We compared the 52 detailed feeder models with the larger 3000 feeder database with regards to feeder capacity margin and feeder PV penetration (Fig. 18). It can be seen that the feeder models provided by PG&E are representative of the larger feeder population.

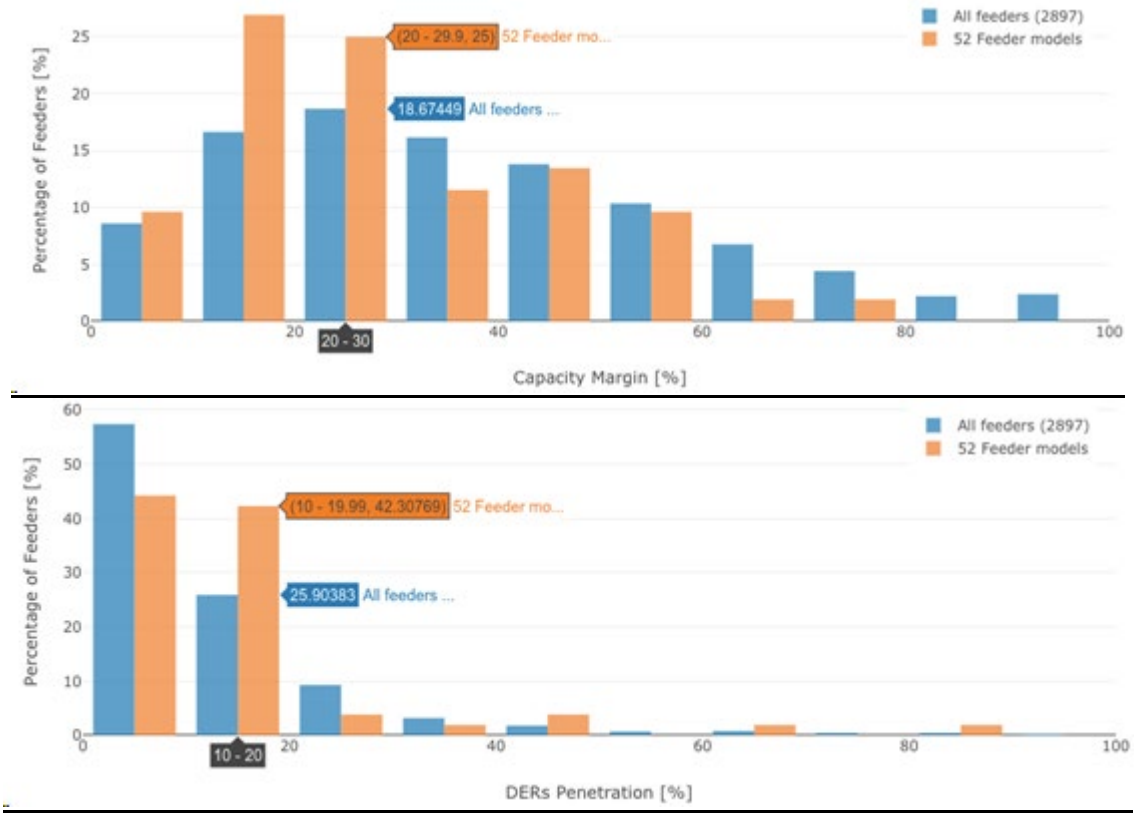

Figure 18: Capacity margin (top) and DER penetration in percentage (bottom) for more than 3000 PG&E feeders (orange) as well as for the 52 detailed feeder models (blue).

Smart inverter capabilities as FMUs

CyDER supports the following types of simulations related to PV integration:

- PVs without any smart inverter control.
- With Volt/Var control only.

- With Volt/Var/Watt control (reactive power support and active power curtailment).
- With Volt/Var/Watt control and additionally operation of a micro-storage embedded in the inverter to manage curtailed PV active power.

For the Volt/Var curve various settings have been considered in our simulations, which reflect settings used by some utilities. Specifically, this includes:

- Base case: PG&E's curve saturating at 30% of reactive power (left plot of Fig. 19).
- HECO's curve saturating at 44% of reactive power (right plot of Fig. 19).
- Aggressive version of HECO's curve saturating at 60% of reactive power (the curve is similar to the one in Fig. 19, but with a different saturation point).

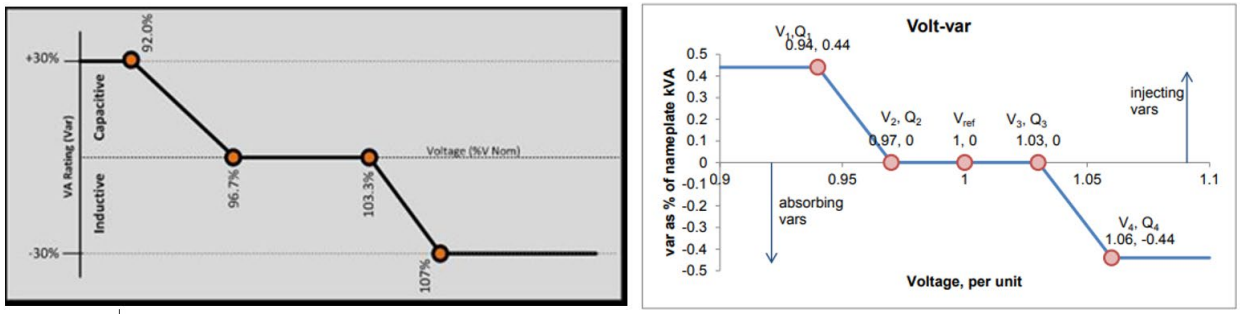

Figure 19: PG&E's (left) and HECO's (right) Volt/Var curves.

For the Volt/Var/Watt controller we envision a micro-storage (e.g. super-capacitor or battery) to be an integral part of the inverter with the goal of locally storing the curtailed PV energy, if capacity is available, instead of discarding it. The objective of the micro-storage is to absorb PV generation during generation peak times, and deploy it at later times in the day (i.e. peak load times). This approach is interesting for future grids with large PV penetration and is particularly relevant for PV installations located far away from the feeder head since they are curtailed more often than those close to it. The integrated micro-storage can help mitigate current (i.e. duck curve) and future (i.e. energy imbalance, ramping, voltage issues, etc.) challenges on the power grid. Another application could be the under-sizing of PV inverter, to allow a bigger PV array to be connected to the same size inverter (thereby increasing the utilization of the inverter and lowering the investment cost). Excess energy which cannot be exported due to the inverter power limit is stored in the micro-storage, and deployed later in time when capacity is available.

Whereas multiple options exist for controlling the micro-storage, the simple control logic of Fig. 20 was adopted in this project. The micro-storage is charged with $P_{charging}$ when PV curtailment occurs, and capacity is available. Once the PV curtailment ends, the storage would start to discharge, if the following requirements are satisfied: (a) the local voltage is below a defined threshold (i.e. below 1.02 p.u.), (b) energy from the PV system was actually stored in the micro-storage (i.e. avoid discharging of intentionally stored energy for back power), and (c) the inverter power limit is not exceeded (Note: the power limit is user-configurable with the parameter *a* as a scaler of $P\_{nominal}$). Fig. 21 illustrates the functionality of the FMU.

Charging management

if ($P_{curtail}$ > 0) and (SOC < SOC_{max}) and ($P_{charging}$ < $P_{nominal}$):

$P_{charging} = \min(P_{charging} + P_{curtail}, P_{nominal})$

if (SOC >= SOC_{max}):

$P_{charging} = 0$

$E_{curtail} = \text{Integral}(P_{curtail})$

Discharging management

if (Voltage < $V_{lower\_threshold}$) and (SOC > SOC_{min}) and ($P_{discharging}$ < $P_{nominal}$) and ($E_{curtail}$ > 0):

$P_{discharging} = \min(P_{discharging} + a^* P_{nominal}, P_{nominal})$

if (SOC <= SOC_{min}) or (Voltage > $V_{upper\_threshold}$):

$P_{discharging} = 0$

Figure 20: Charging and discharging algorithm for the embedded micro-storage.

Parameters

Type	Name	Default	Description
String	weather_file	""	Path to weather file
Real	n	26	Number of PV modules [1]
Real	A	1.65	Net surface area per module [m2]
Real	eta	0.158	Module conversion efficiency [1]
Real	lat	37.9	Latitude [deg]
Real	til	10	Surface tilt [deg]
Real	azi	0	Surface azimuth 0-S, 90-W, 180-N, 270-E [deg]
Real	thrP	0.05	P: over/undervoltage threshold
Real	hysP	0.04	P: Hysteresis
Real	thrQ	0.04	Q: over/undervoltage threshold
Real	hysQ	0.01	Q: Hysteresis
Real	SMax	7.6	Maximal Apparent Power [kVA]
Real	QMaxInd	3.3	Maximal Reactive Power (Inductive) [kvar]
Real	QMaxCap	3.3	Maximal Reactive Power (Capacitive) [kvar]
Real	Tfirstorder	1	Time constant of first order [s]
Real	P_discharge	1	Battery Maximal Discharge [kW]
Real	EMax	1	Battery Capacity [kWh]
Real	SOC_start	0.1	Initial SOC value [1]
Real	SOC_min	0.1	Minimum SOC value [1]
Real	SOC_max	1	Maximum SOC value [1]
Real	etaCha	0.96	Charging efficiency [1]
Real	etaDis	0.96	Discharging efficiency [1]

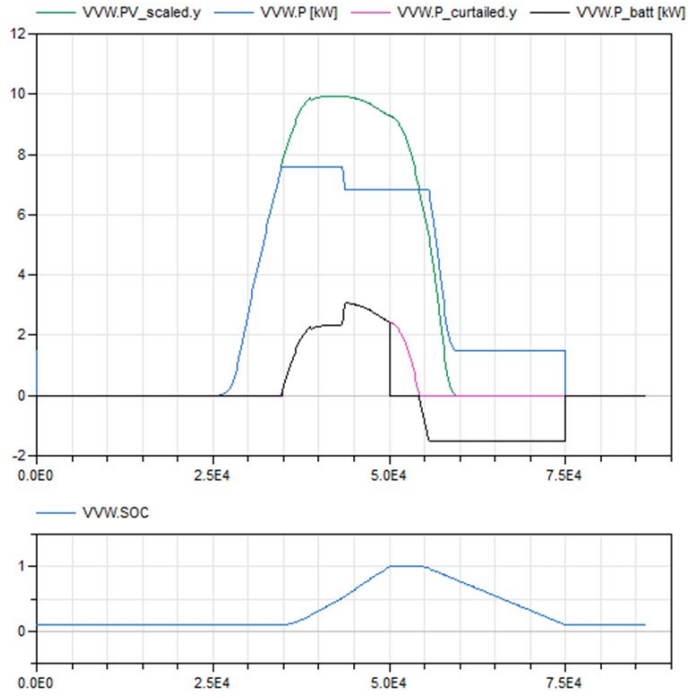

Figure 21: Parameters of the Volt/Var/Watt/Watt FMU (left) and simulation results (right).

The table on the left of Fig. 21 lists all parameters available to configure the Volt/Var/Watt/Watt FMU, and their default values. The plot on the right shows the simulated results for a single day of high PV generation. The base PV generation is shown in green. For the morning hours, the inverter output, in blue, tracks the PV generation. At around 10 AM the inverter reaches its output limit of 7.6 kVA and excessive PV generation is set to be curtailed. The curtailment is shown in pink. However, the micro-storage is

available, shown by a low State of Charge (SOC) in the lower plot in blue, and instead of actually curtailing the PV generation, the micro-storage is charged instead, shown in black. Towards the end of the day, the micro-storage reaches its maximum SOC and stops charging, indicated by the battery power, in black, dropping to 0 W. Actual PV curtailment takes place during this time. Once the PV generation decreases below the inverter power limit, the control logic of the micro-storage starts to discharge, to fill the gap between PV generation and inverter power limit, until it reaches the defined maximum discharge rate of 1.5 kW, shown by the negative portion of the black line. The SOC is decreasing, shown in the lower plot, and discharging stops when the micro-storage reaches its empty level.

Long-term load and PV scenarios

PG&E provided LBNL with long-term forecasts of peak load demand (total consumption at the feeder head) and PV capacity for the period 2019 - 2027 for each of the 52 feeder models. Fig. 22 shows the load forecast for each of the substations considered in the project. Using this data, we computed the forecasted PV penetration for each year (in %) by dividing the installed PV capacity by the peak load demand. Fig. 23 shows the projected PV penetration of each feeder for the year 2027. The average PV penetration is 12% and the median is 9.8%, but there are also feeders with much lower or higher PV penetrations. For example, feeder BRITTON 1115 has a forecasted PV penetration of only 3% in 2027, whereas the expected PV penetration for feeder BURLINGAME 0405 is 67% for the same year. Overall, the forecasted PV penetration for these 52 feeders in the PG&E territory is rather low.

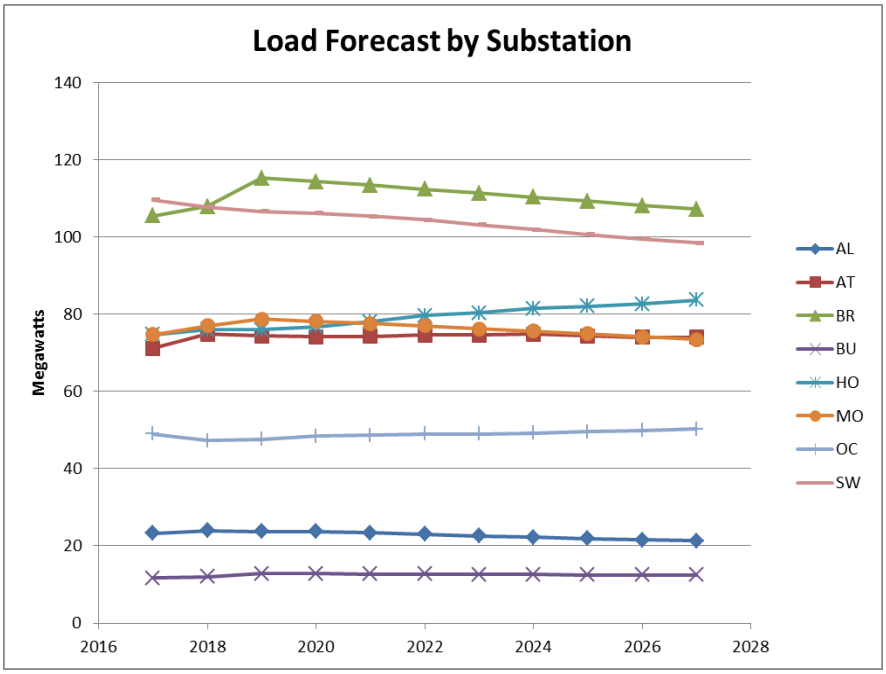

Figure 22: Long-term load demand forecasts for each of the considered substations.

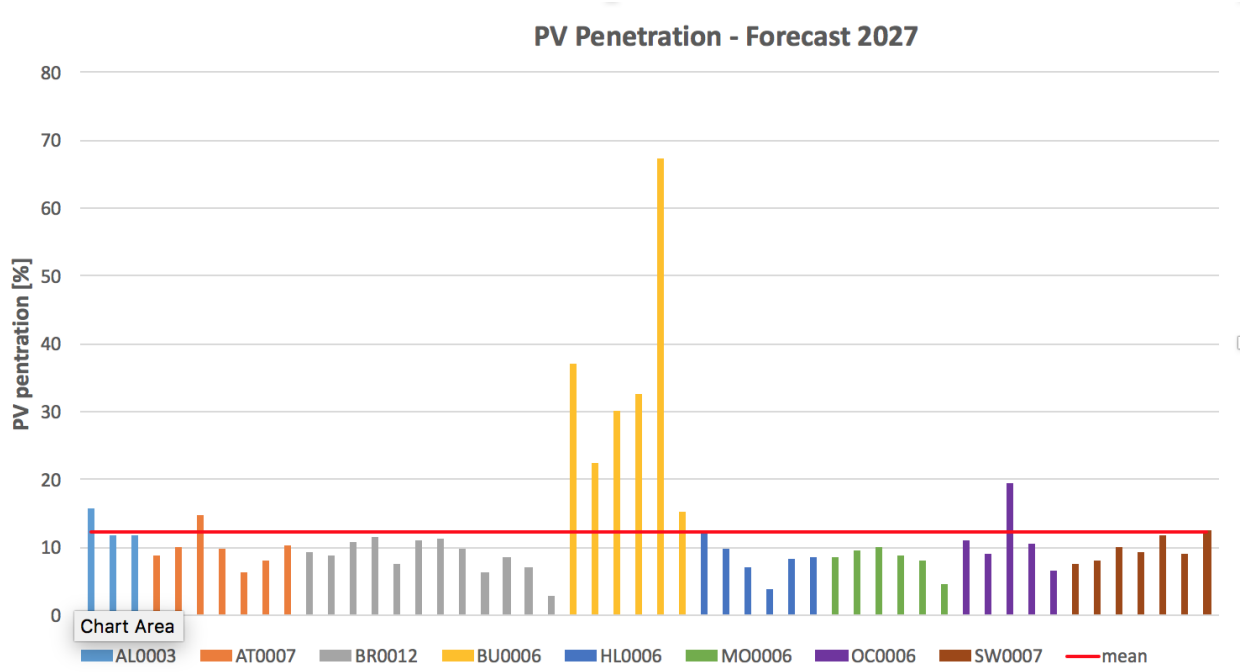

Figure 23: PV penetration forecast for the year 2027 for 52 feeders.

Simulation results with PG&E scenarios

We simulated each of the 52 PG&E feeders in two different case studies. First, we simulated for every year from 2019 to 2027 with the PV and load forecasts provided by PG&E, assuming that the PV systems are connected in the second half of each feeder. This analysis investigates the impact of forecasted installed PV capacity on the grid. Second, by fixing the PV penetration to the forecast of 2027, we simulated the grid with different distributions of the PV systems across the network nodes. This analysis focuses on the effect of the spatial PV distribution for the same overall PV capacity. We decided to select the most critical day of the year (the day with the largest imbalance between power consumption and generation) in 2027 and run all simulations for 2019 to 2027 for the same calendar day. We ran the simulations for the three different inverter control scenarios: no control, Volt/Var control and Volt/Var/Watt control. These results were included in a separate report submitted to PG&E including the raw simulation results. The rest of this section will present indicative results.

As depicted in Fig. 24, we created an interactive Excel spreadsheet which provides a visualization of the results and input parameters (including forecasts) for simulations across multiple feeders. Fig. 25 presents the number of feeders with Volt/Var and Volt/Watt control activation for each of the years in the forecast horizon. In 2019, only 13 feeders showed an effective Volt/Var control, whereas 2 of these feeders additionally applied Volt/Watt control due to significant overvoltage issues. The number of feeders where Volt/Var control was activated increased to 23 by 2027. On the other hand, the number of feeders where a Volt/Var/Watt control had an additional impact remained constant at 2 throughout the simulated time horizon. These results showcase the necessity of Volt/Var control with increasing PV penetration in the PG&E feeders. Nevertheless, with the exception of some special cases, active power curtailment is not expected to be necessary for these feeders within the next 9 years. The left plot of Fig.

26 shows the worst-case voltage profile within the forecast horizon for a feeder where only reactive power control is activated, whereas the right plot shows the same result for a feeder where active power curtailment is needed in addition.

For the PV distribution analysis, we consider a scenario with the PVs spread evenly throughout the feeder, as well as scenarios where they are allocated in the last 75%, 50% and 25% of the feeder length. Fig. 27 shows how the maximum voltage depends on the PV allocation for each of the three control scenarios. As expected, the further away from the feeder head the PV systems are located, the larger the benefit of smart inverter controls. Note that as this was a worst-case analysis, PVs were not correlated with the load peak demand at each node. Nevertheless, this correlation is included in the next section "Simulation results with PV penetration up to 200%", namely PV location and installed power depend on the peak load demand on each node.

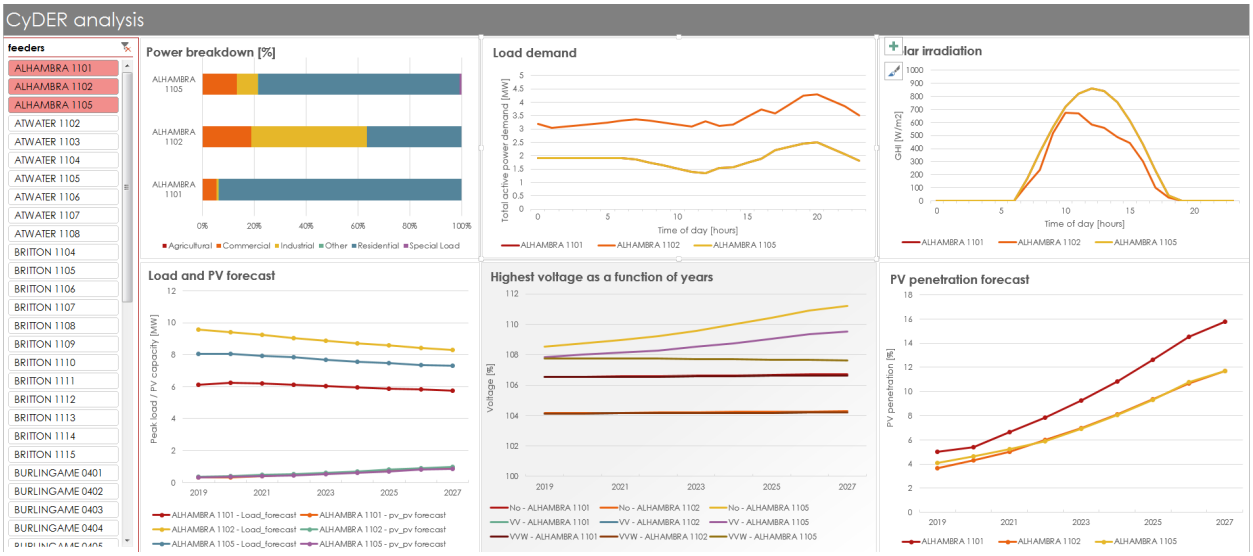

Figure 24: Dashboard in Excel that visualizes results for any combination of the feeders.

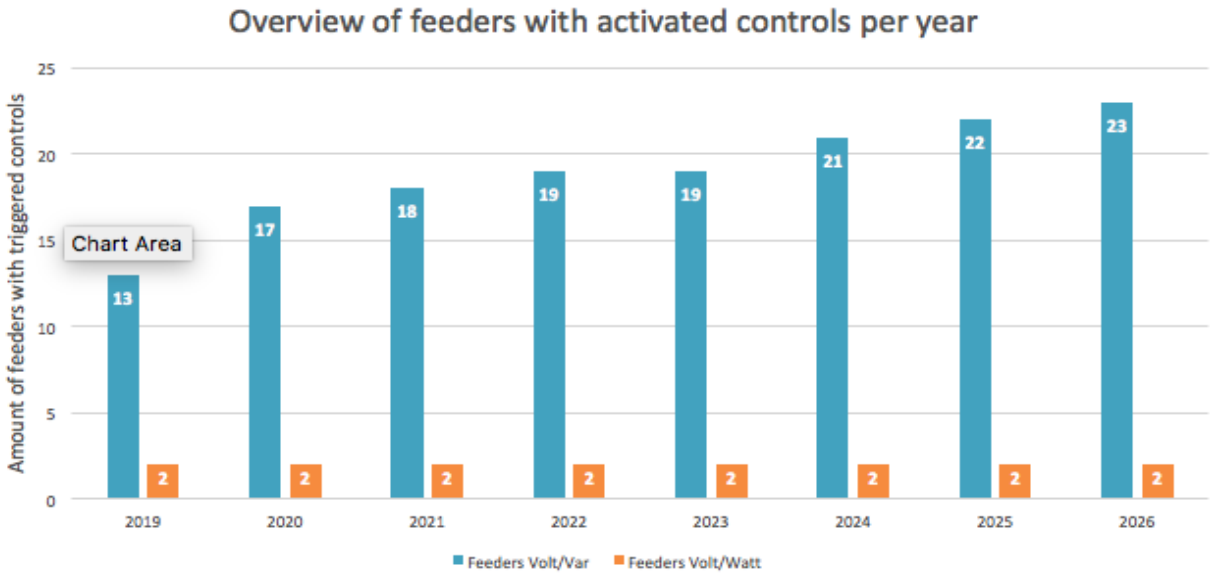

Figure 25: Overview of the effectiveness of Volt/Var and Volt/Var/Watt controls.

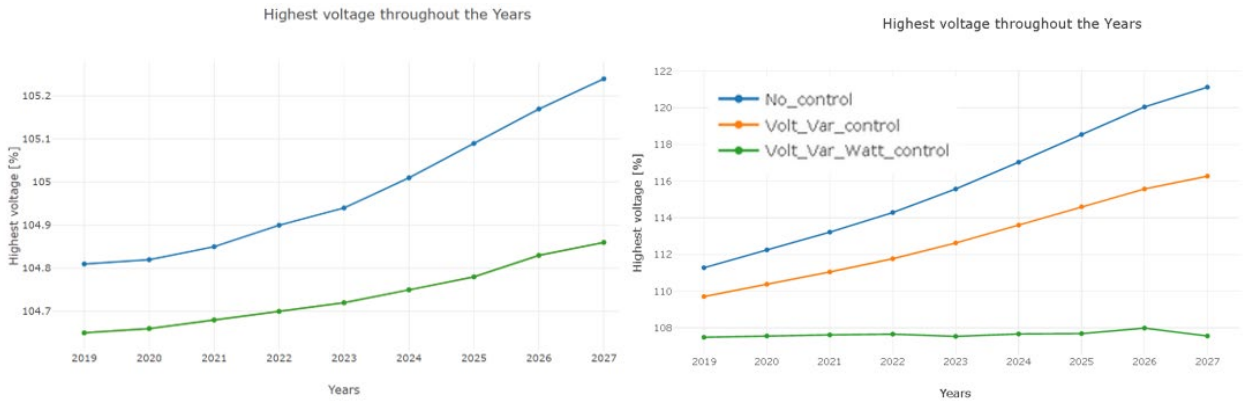

Figure 26: Left: Feeder BURLINGAME 0404 with only Volt/Var control activation throughout the forecast horizon. Right: Feeder SWIFT 2110 *with all three controllers activated.*

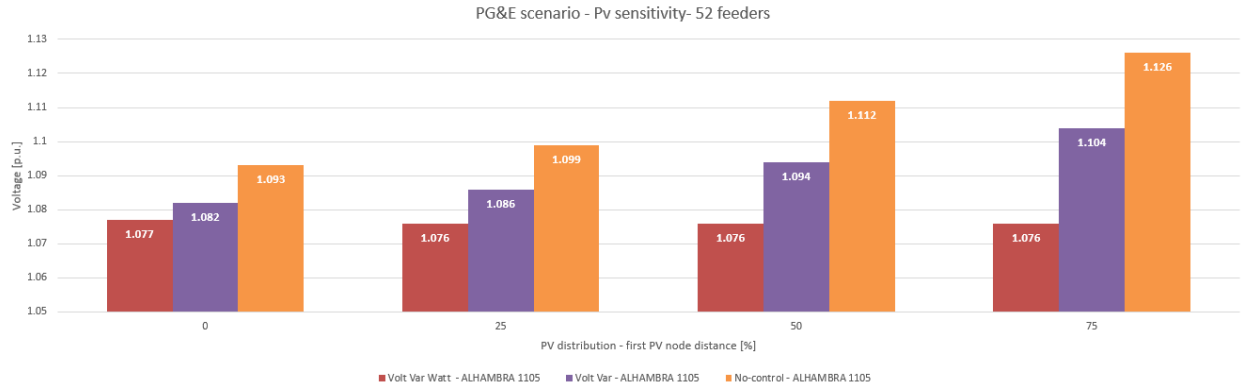

Figure 27: PV distribution (allocation) analysis for ALHAMBRA 1105.

Simulation results with PV penetration up to 200%

Fig. 28 illustrates an example of the capabilities of the model exchange simulation mode. Due to the better handling of oscillations caused by feedback loops, it is able to successfully simulate scenarios with extreme PV penetration. The plot shows the results for the feeders BR1106 (left plot) and BU0405 (right plot) for PV penetrations in the range 0-200% and 0-250%, respectively, for 3 different controls. For the BR1106 feeder, 10 equally sized PV systems are installed at the locations of the 10 largest customers of the second half of the feeder. The difference for feeder BU0405 is that 25 PV systems are installed at the locations of the largest 25 customers. It can be seen, that especially the Volt/Var/Watt control can be used to keep the voltages below a set threshold even for scenarios with very high PV penetration.

In general, the amount of iterations of a simulation increases with the amount of FMUs and especially with the amount of existing feedback (algebraic) loops. While carrying out the simulations with extreme PV penetration, one limitation of the ME master was identified, namely an exponential increase in simulation time per iteration. Preliminary investigations showed that this issue is likely related to some internal functioning of SimulatorToFMU and/or PYFMI and could be interesting and useful to characterize and resolve in future work.

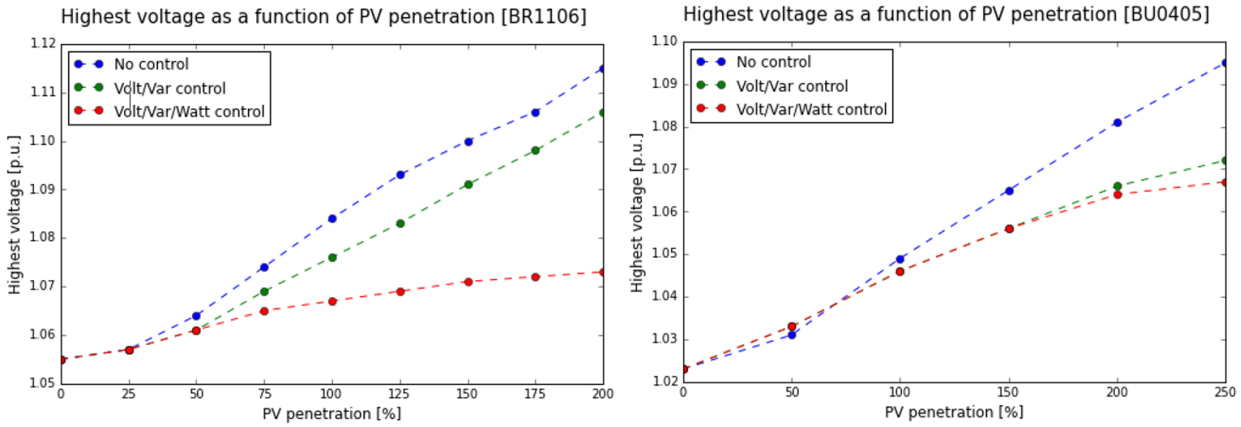

Figure 28: Contribution of Volt/Var and Volt/Var/Watt controls in reducing overvoltages in the case of extreme PV penetration scenarios (up to 200%) for two feeder models.

Simulations with battery control for management of curtailed PV energy

Finalizing the ME master allowed us to simulate scenarios with PV inverters equipped with a micro-storage unit for management of the PV curtailed energy. The micro storage can store energy which would have been curtailed by Volt/Watt control and export it back to the grid when voltages are lower again. The battery operation is according to the algorithm of Fig. 20 and is shown in Fig. 29. An interesting analysis hereby is to characterize the dependence of curtailed PV energy on the battery power rating and energy capacity. For this purpose, we ran simulations with different values of power rating and energy capacity, both defined in relation to the peak power of the installed PV. Fig. 30 plots the lost energy due curtailment over the storage capacity of the battery. The different colored lines represent different power ratings of the micro-storage unit. It can be seen that the effect of increasing storage capacity is limited by the charging power of the battery. It has to be mentioned, that even an oversized battery in terms of capacity and charging power cannot compensate all curtailment losses due to battery losses while charging and discharging.

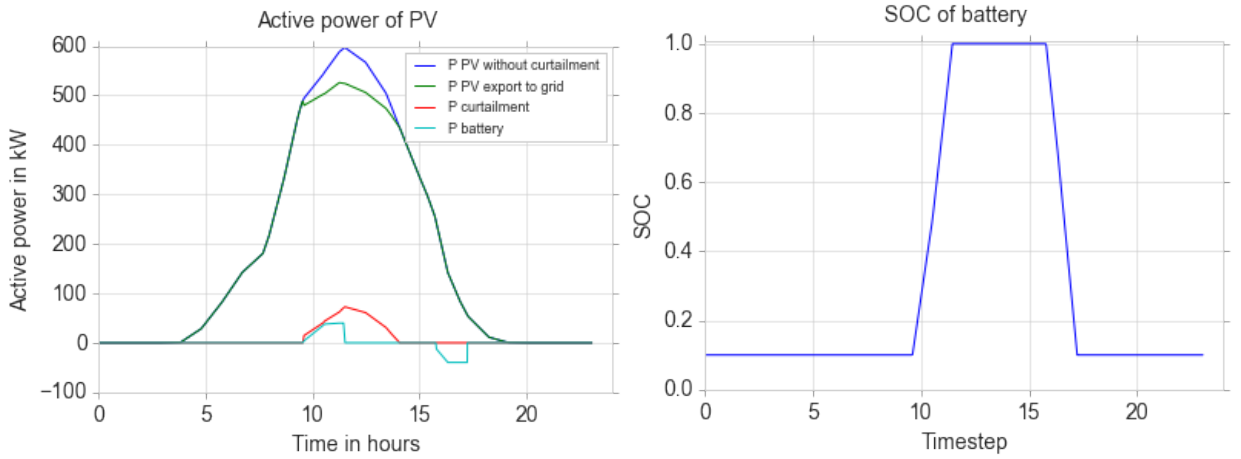

Figure 29: Battery charging and discharging profile (left) and SOC (right).

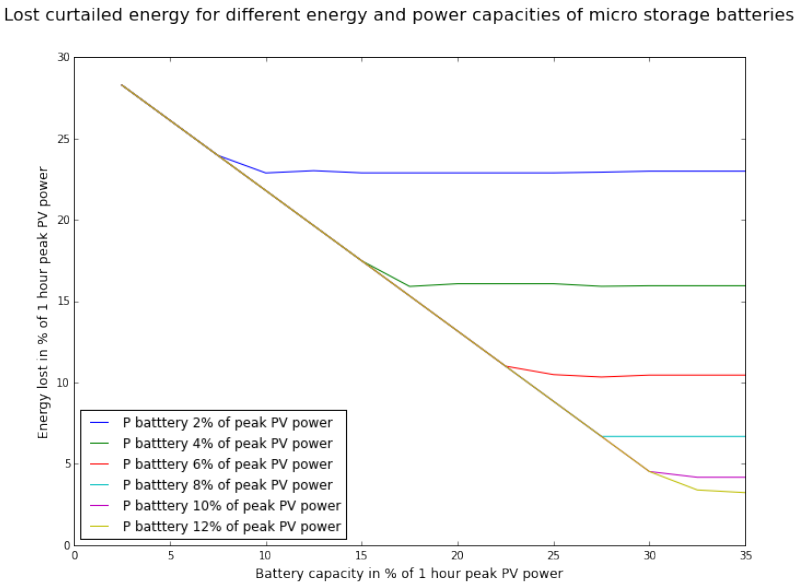

Figure 30: Lost PV energy due to curtailment as a function of the battery energy capacity (x-axis) and the battery power rating (various colored curves).

Supervisory optimizer of Volt/Var/Watt inverter setpoints

A supervisory controller to periodically optimize the setpoints of the Volt/Var/Watt curves of smart inverters was developed within the project. The controller is centralized and we parameterize each inverter's reactive power injection and active power curtailment as *piecewise linear* functions of the voltage at the inverter's bus, which in turn depends on the load demand and PV power output. In more detail, we first consider a number of possible PV output scenarios for each of the PV systems, e.g. [0-100%] of the PV system's rated power with a step of 10%, and run an Optimal Power Flow (OPF) based optimization problem for each of these scenarios. This will generate tuples ($P_{curt,i,p}$, $Q_{PV,i,p}$, $V_{i,p}$) that contain the optimal reactive power injection and active power curtailment for each of the PV nodes (indexed by *i*) for the voltage scenarios corresponding to each of the PV production scenarios (indexed by *p*).

The optimization problem will run every 15 minutes and sequentially update the tuples using the most recent load forecast for each bus for the next 15-min period. At the beginning of each 15-min period, the most recently computed tuples will be sent from the centralized controller to each PV inverter. It is worthwhile to notice that the centralized controller is parametric with respect to the PV production, which is parametrically evaluated instead of explicitly forecasted. This parametric approach spans different scenarios of possible PV power output and allows us to come up with piecewise linear Volt/Var/Watt curves by interpolating the ($P_{curt,i,p}$, $Q_{PV,i,p}$, $V_{i,p}$) tuples. In this way, the real-time Volt/Var/Watt control computes the desired control action by just evaluating the piecewise linear curve, similarly to existing Volt/Var/Watt approaches.

The optimization problem was set up as shown in Fig. 31. The goal is to minimize the utilization of the PV control resources, i.e. the curtailed output power as well as the reactive power consumption, while minimizing network losses and maintaining all bus voltages and branch currents within allowable limits. Slack variables (η) are used to assure feasibility of the optimization problem. A single iteration of the Backward-Forward Sweep (BFS) method is used to model the distribution network, whose structure is

captured by two matrices: The Bus Injection to Branch Current (BIBC) matrix (includes only ones and zeros) and the Branch Current to Bus Voltage (BCBV) matrix (includes line impedances). The advantage of using this power flow model is its linearity, which keep the overall problem a convex optimization problem. Constraints (1f) reflect the physical limitations of the converter in terms of its output power, whereas constraints (1g) incorporate the trivial fact that it is not possible to curtail more PV power than it is produced. Considering that only a single iteration of the BFS procedure is used, the OPF solution is not exact and an additional iterative procedure is needed to improve the algorithm's performance. More precisely, once $P_{curt,i,p}$ and $Q_{PV,i,p}$ are computed after each iteration, they are used as input within a BFS-based power flow to compute the resulting voltages that are then used as V_i^* in (1b) in the next iteration. This process continues until the voltages converge and then a near-optimal OPF solution is obtained.

$$\min_{\boldsymbol{P}^{\text{curt}},\boldsymbol{Q}^{\text{PV}}} \quad \sum_{i\in\mathcal{N}_n}(c_i^p P_i^{\text{curt}} + c_i^q Q_i^{\text{PV}}) + \sum_{j\in\mathcal{N}_b}(c_j^l P_{\text{loss},j}) + \|\eta_V\|_\infty + \|\eta_I\|_\infty \tag{1a}$$

$$\text{s.t} \quad I_i^{\text{inj}} = \frac{(P_i^{\text{inj}} + Q_i^{\text{inj}})^*}{V_i^*} \qquad \forall i \tag{1b}$$

$$I_b = BIBC \cdot I^{\text{inj}}, \tag{1c}$$

$$\Delta V = BCBV \cdot I_b \tag{1d}$$

$$V_i = V_{\text{slack}} + \Delta V_i, \qquad \forall i \tag{1e}$$

$$(P_i^{\text{PV}} - P_i^{\text{curt}})^2 + (Q_i^{\text{PV}})^2 \le (S_i^{\text{nom}})^2, \qquad \forall i \tag{1f}$$

$$0 \le P_i^{\text{curt}} \le P_i^{\text{PV}}, \qquad \forall i \tag{1g}$$

$$V^{\min} - \eta_{V,i} \le |V_i| \le V^{max} + \eta_{V,i}, \qquad \forall i \tag{1h}$$

$$I^{\min} - \eta_{I,i} \le |I_i| \le I^{max} + \eta_{I,i}, \qquad \forall i \tag{1i}$$

Fig. 31: The optimization problem formulation of the supervisory Volt/Var/Watt controller.

Fig. 32 shows simulation results with the supervisory controller using a 19-bus CIGRE radial network with four PV systems connected to buses 12, 16, 18 and 19. Due to limited current flow of the line to bus 16, a significant amount of active power curtailment is required. The inverters at the remaining three PV buses contribute with reactive power control only. Note that the optimal result is not necessarily a linear curve, which showcases the value of having an online optimization scheme. In the remaining project time during the no-cost extension, we plan to integrate the supervisory controller in CyDER and repeat simulations including the micro-storage controller.

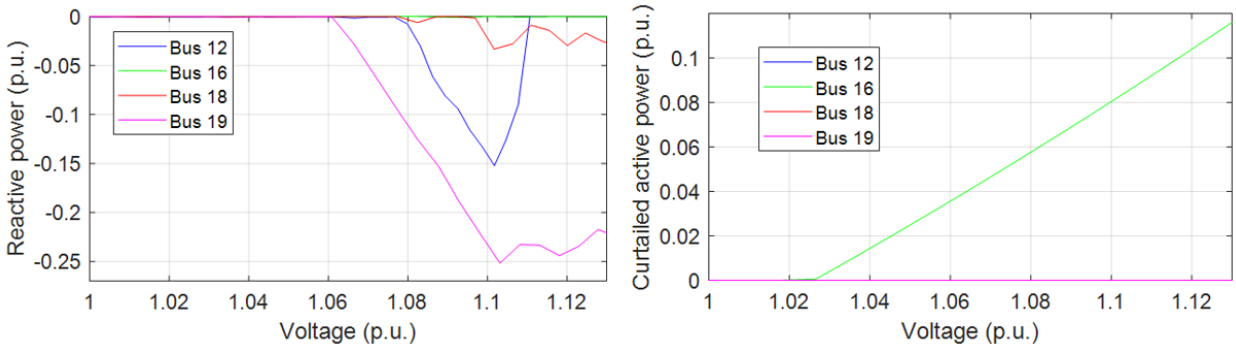

Fig. 32: Results from optimizing the settings of Volt/Var (left) and Volt/Watt curves (right) in the 19-bus CIGRE grid (hard voltages limits were fixed to 0.9 pu and 1.1 pu).

Impact of EV charging on distribution grids

A separate line of work in this task was centered around the impact of uncontrolled EV charging on distribution grids, in collaboration with PG&E and the National Resources Defence Council (NRDC) [20]. Using the substation/feeder models and SCADA data provided by PG&E, we ran power flow simulations assuming a 100% EV penetration rate (defined as one EV per residential customer connection) and identified the effect on remaining capacity at the feeder head, worst-case line loading, and worst-case undervoltage. Illustrative results for 39 feeders ordered by the ratio of residential energy demand are shown in Fig. 33. Based on these results, 58% of the feeders exceed their remaining power capacity, 16% are below the 0.95 p.u. voltage limit, and 47% have some line overloading, which indicates that line overloading is more likely the limiting factor.

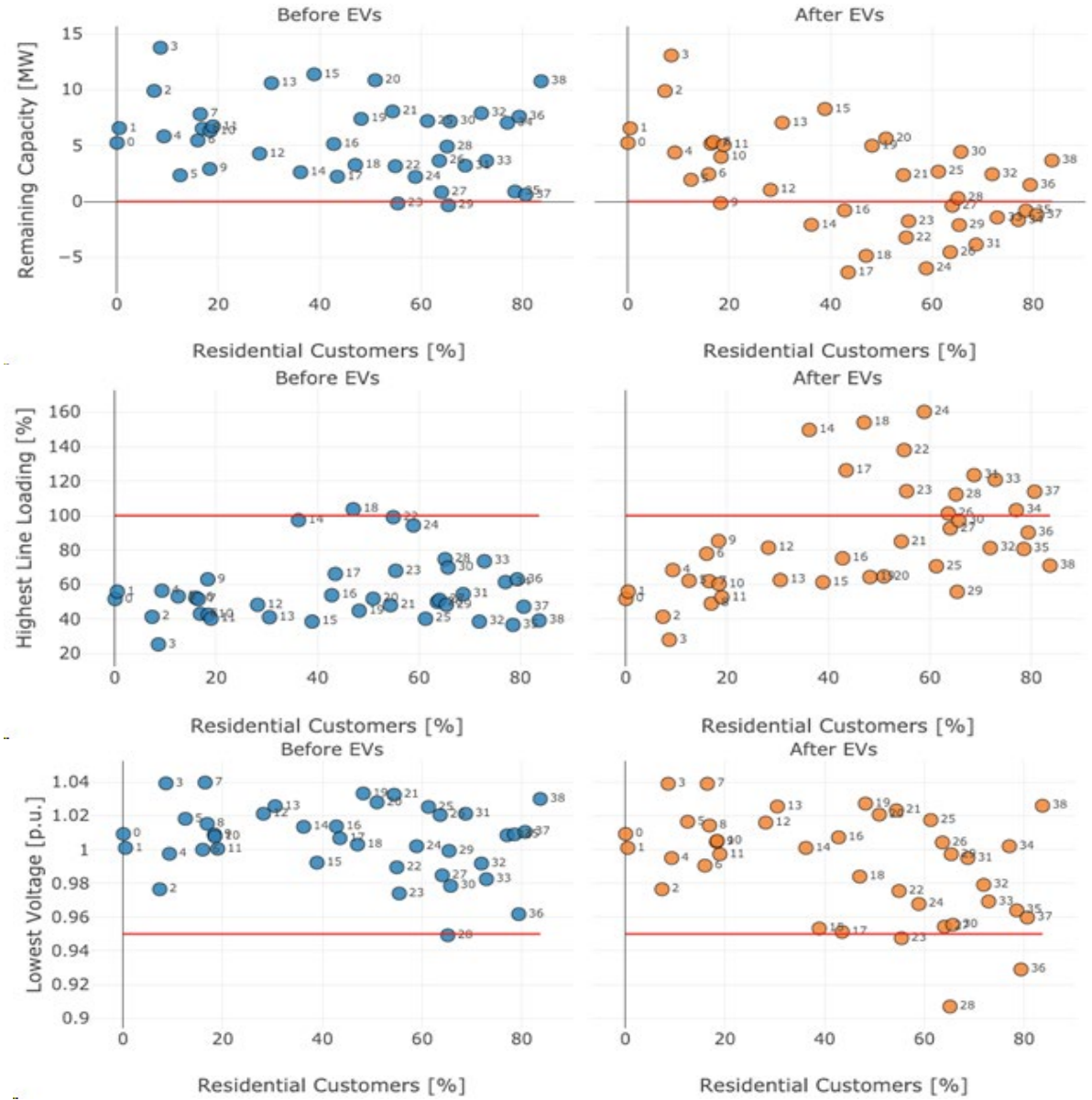

Figure 33: Effect on remaining capacity (top), line loading (middle), and lowest nodal voltage (bottom). Left: feeders without EV demand. Right: feeders after adding one EV per household.

In a second step, we attempted to extrapolate the observations from the detailed simulations on the 39 feeders to a larger dataset consisting of 1054 feeders in the San Francisco Bay Area. This analysis revealed that approximately 60% of the feeders in the San Francisco Bay Area would reach or exceed their maximum loading limit, and therefore Time of Use (TOU) tariffs or other control mechanisms would likely be required. Interestingly, additional investigations revealed that, even with perfect EV control and communication, roughly 60% of the feeders would still experience peak demand increase at 100% EV penetration scenario compared with today's situation. Nonetheless, EV control would limit the peak increase to 8% of the current peak demand on average, compared with a 64% average increase in the uncontrolled case.

Task 3.2: HIL testing

Preparation work

Ametek recommissioning: This task focused on the main HIL testing following up on the preliminary work in BP2. As mentioned earlier, the Ametek amplifier failed in BP3 and was replaced by a new device in BP3. Therefore, the first tasks hereby were to re-energize Ametek, connect it with the OPAL-RT simulator, re-calibrate the DC offsets, testing of safety measures, designing a PI controller to stabilize the Ametek output voltage, and developing an automated feature to report errors and log faults.

Reactive power imbalance: The FLEXGRID system is built to represent a residential split-phase system, which is duplicated three times. Hereby each of the three single-phase inverters is connected to a 208V/240V transformer, which are then connected in delta on the primary feed (Ametek or LBNL grid side). The problem originates in the different sizes and location of the PV modules, which cause an imbalance on the system. This imbalance leads to shifted voltage potentials due to neutral point shifting, and ultimately to shifted reactive power profiles, as shown in Fig. 34. In this figure, the reactive power for the three phases is shown over the course of one day. In this setup, the inverters are controlled with a constant power factor of 1 and the only sink of reactive power are the transformers. It can be seen that all phases start with a constant reactive power of about 350 var during the night. Once PV starts to generate, one phase continues to show a constant reactive power, while the other phases peak with reactive powers of 1000 var and -200 var.

The CyDER team discussed the issue and agreed on virtually symmetrizing the system by averaging the reactive power across the three phases. This averaged reactive power values is then used for each phase and sent to the real-time simulation model. In the example shown, the average of 1000 var and -200 var would be 400 var per phase, which is the expected true value of reactive power. Assuming the system to be fully balanced is also valid considering the quality of the PG&E feeder models and the common practice of PG&E run balanced power flows.

PG&E feeder model assignment: The distribution feeder models used for CyDER are sourced from PG&E in the CYME format. They are converted to ePhasorsim (the model environment within the real-time simulator Opal-RT), and a dynamic load demand is assigned to loads along the feeder based on PG&E SCADA data. Following the standard practice of PG&E, a balanced power flow simulation is run in CYME where three-

phase loads are balanced, while single-phase and two-phase loads are first internally converted to three-phase and subsequently balanced. However, when the CYME models are converted to ePhasorsim, the basic configuration of the loads, e.g., single-phase loads, remains. As a result, it is not possible to a-posteriori balance single-phase loads in ePhasorsim because phases B and C do not even exist in the model. In order to compensate for this inconsistency, the real-time simulation model was virtually symmetrized by ignoring single-phase and two-phase loads and using only phase A. Specifically, the power and voltage of phase A are duplicated for phases B and C within the real-time simulation model to represent a balanced system.

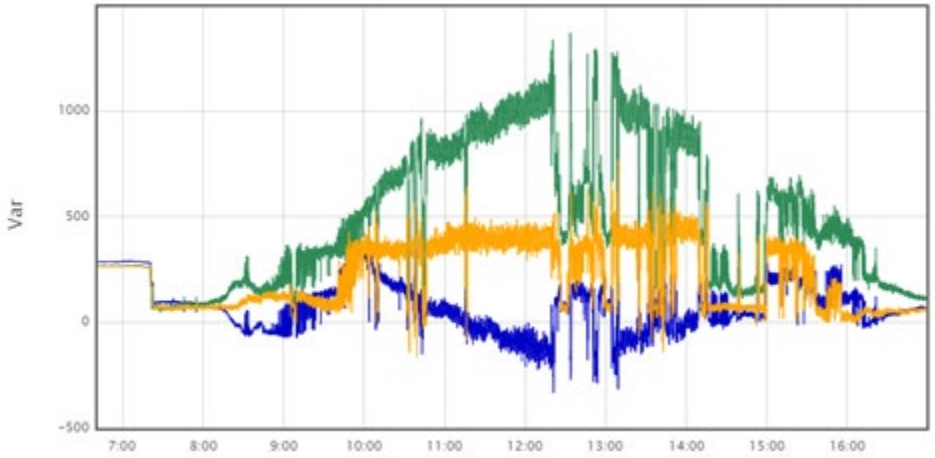

Figure 34: Imbalance of reactive power at FLEXGRID (each color corresponds to a different phase). The measurements are taken at the primary (delta) side of the transformer.

OPAL-RT API calls: During the first HIL tests in BP2, the API to the real-time simulation model (through OPAL-RT/RT-Lab) caused problems when invoked too often. Some testing revealed that after about 250 calls, the API would fail, causing all recorded data to be lost and a stop of the experiment. With a simulation time step of 5 minutes, this limited number of API calls restricted testing for up to approximately 24 hours. In BP3 a new RT-Lab version (11.3.6) was available, but was found to still have some limitations as API calls fail somewhere between 2 and 600 calls.

To resolve this issue, the CyDER team redesigned the Real-Time Simulator FMU, which encapsulates the OPAL-RT API, to have two separate functionalities. First, the OPAL-RT model is automatically compiled and launched using the RT-Lab API. Second, instead of exchanging information (i.e. set P and Q of simulated PV, get nodal voltages) through the RT-Lab API, the team encapsulated a MODBUS communication which communicates directly with the real-time simulation model. This avoids the involvement of RT-Lab and allowed API calls for several days using a 20 second time step (maximum tests were about 15,000 API calls).

Volt/Var/Watt control - Repetition of BP2 testing in closed loop

The first test was basically a repetition of the test ran in BP2 but in closed loop, i.e. with the Ametek grid emulator energized. With reference to Fig. 13, the simulated PV systems are about 900 kW peak each (total 2.7 MW), and are oriented towards the south, west, and east. Each system can provide reactive power of 70 kvar (210 kvar total). The

real PV system at FLEXGRID is scaled from its nominal size of 15 kW to 900 kW peak to match the simulated PVs. Similarly to the BP2 setup, the simulated PV systems provided Volt/Var control, whereas the real PV system Volt/Watt control. The test was successful and demonstrated CyDER implementation in a closed-loop HIL environment. The results obtained from the HIL simulation were compared against the results obtained from a CYMDIST simulation, as shown in Fig. 35. It can be seen that the RMSE is 1.6% in the worst case which is well below the goal from the SOPO (maximum 5% discrepancy).

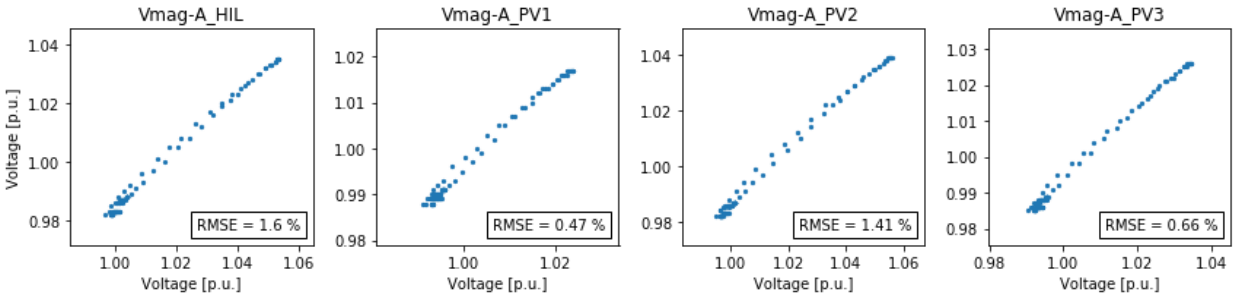

Figure 35: Comparison of voltages from the HIL and CYMDIST simulations.

Volt/Var/Watt control + micro-storage

The second test was focused on applying a combined Volt/Var/Watt control at the real PV system, as well as exploiting the Tesla Powerwall batteries to manage the curtailed PV energy (according to the logic in Figs. 20 and 21). Experimental results with this setup are shown in Fig. 36. The control action of the Volt/Var/Watt/Watt controller is visible in the second plot. During the first half of the day, the battery power, in brown, increases as the local voltage increases due to raising PV generation. At around noon the battery reaches its full state of charge which causes the system voltage to steeply increase, and the PV curtailment, in blue, to start. The curtailment reaches a maximum of 60%, and then decreases as PV production decreases. During night time, the battery, in brown, is discharged as system voltage decreases. The test demonstrates the successful operation of a simple rule-based control for batter dispatch management, but it also illustrates potential limitations related to sharp voltage increases when the battery reaches full charge. Such sharp increases could be mitigated with the application of predictive battery control strategies.

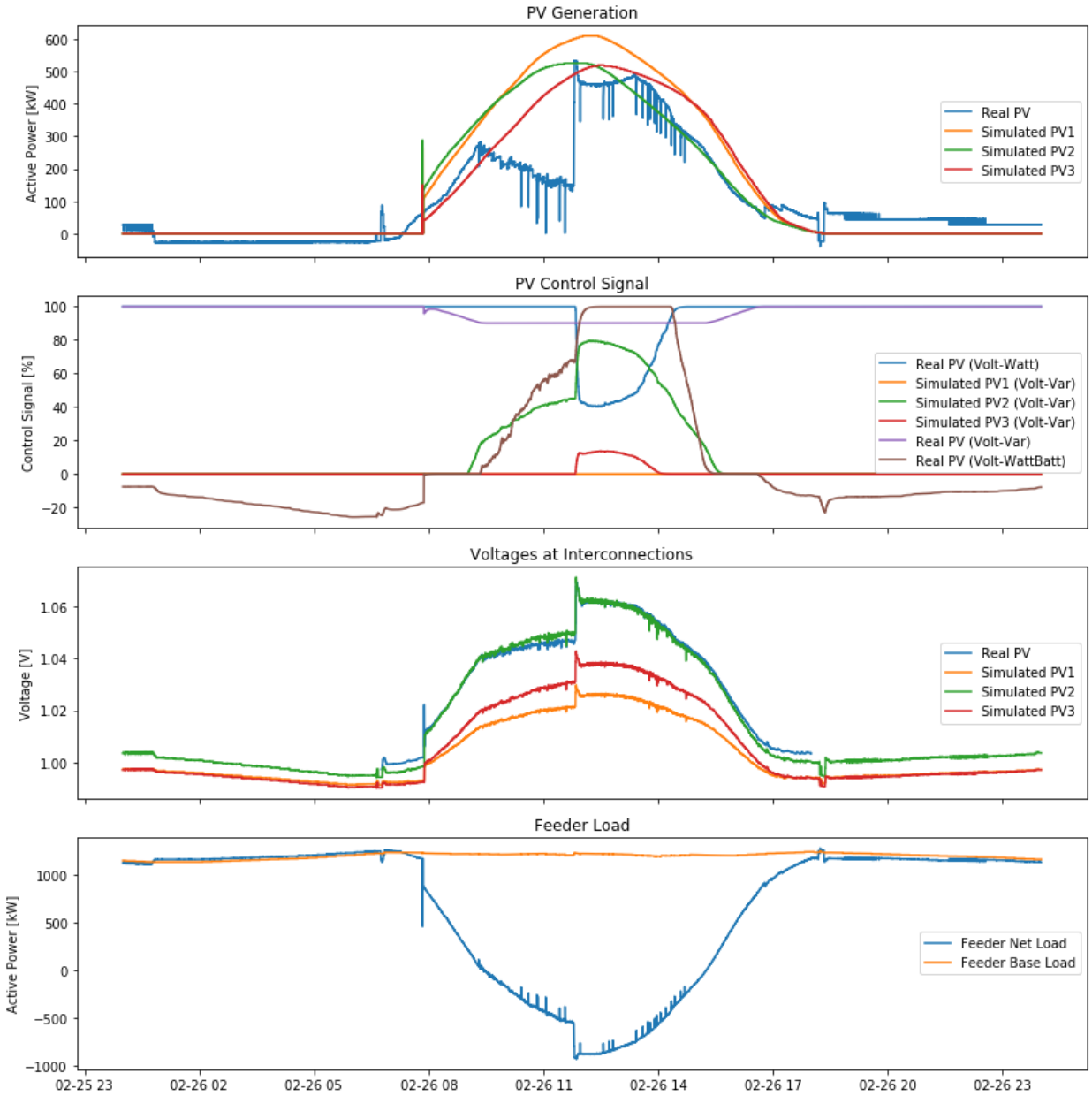

Figure 36: HIL result for the Volt/Var/Watt/Watt test (including battery storage).

Volt/Var/Watt control in combination with MPC

The final HIL test conducted at the FLEXGRID facility involved the replacement of the real PV controller. Whereas the PV inverter was configured with a Volt/Var/Watt controller similarly to earlier tests, the batteries were controlled with a Model Predictive Controller (MPC). This controller was developed for a synergistic California Energy Commission (CEC) funded project by LBNL, and branded Distributed Optimal and Predictive Energy Resources (DOPER). DOPER's objective is to minimize the total energy cost for a customer, taking Time of Use electricity rates into account. It internally forecasts PV generation and building load to compute the optimal battery dispatch. The MPC prediction horizon was set to 24 hours with a 5-minute control time step.

The development of the necessary wrapper to wrap DOPER as an FMU led to the issue of useable memory with 32 bit applications. With a 32-bit application the system can only address roughly 3.5 GB of system memory (RAM) for an application. While this limit was previously sufficient for CyDER to run FMUs, the DOPER FMU requires more memory which led to memory errors and eventually resulted in tests involving MPC stopping after a certain number of iterations. In order to avoid this issue, the CyDER team worked on a migration of the SimulatorToFMU tool to allow the export of 64 bit FMUs, which enables using all available RAM of modern PCs. Besides exporting DOPER as an FMU, this modification of SimulatorToFMU will be useful in future large-scale, memory-intensive applications of CyDER such as electromagnetic transient simulations of low-inertia systems and high-fidelity, multi-energy city, district or state-wide simulations.

Using the upgraded SimulatorToFMU, the plan was to convert all involved FMUs to the 64-bit version including the new orchestrator (master script), PV, controls, micro-PMU, inverter, and OPAL-RT FMU. However, the problem is that RT-Lab (the software to interface with the real-time simulator from OPAL-RT) does not support 64 bit. A workaround was developed where FMUs were first exported as 32 bit to start the HIL test. Once initialized, the orchestrator is stopped and the FMUs are exported as 64 bit. An added flag within the OPAL-RT FMU detects the change and stops the communication with RT-Lab. From this point on, all FMUs run in 64 bit, which allows long run times.

The HIL test was conducted for a total of 4.5 days, starting from July 3rd evening, to July 8th noon. The weather conditions were sunny and typical for Berkeley, CA in summer. Note that due to a lack of building load forecast, the HIL test was conducted with a fixed load profile which would represent a perfect forecast. The results below are cropped to 4 full days, illustrating the control of DOPER and impact on the feeder model. The real PV system and the three simulated PV systems (oriented south, west, and east) were scaled to 400 kW. The HIL experimental results are presented in Fig. 37.

The first plot shows the solar radiation. As it can be seen, in the mornings the diffuse horizontal irradiance, in orange, closely follows the global horizontal irradiance, in blue, which indicates overcast morning conditions. The second plot shows the PV active power production. Note that the PV power production of the real system is post-calculated based on the combined power measurement at the inverter output and the power measurements at the battery. Due to the fact that these measurements are not perfectly synchronized, the resulting PV power profile is spiky (blue curve). In contrast, the simulated PVs use the 15-minute averaged solar data as input and therefore the respective curves (orange, green and red).

The third plot shows the control activation. In this experiment, only the Volt/Var controller of the simulated PV2 system gets activated, which experiences the highest voltages (see the fifth plot). The real PV system does not provide Volt/Var control but controls the battery in a way that reduces peak demand and associated costs. The brown curve in the third plot shows the battery dispatch profile (positive values correspond to charging and negative to discharging). The battery charges in the morning hours and at night to cover the power demand in the afternoon when the price is higher. This results in a controlled net load profile without positive peaks in power demand (orange curve) in contrast to building base load without PV or storage (blue curve). The last plot shows the feeder base load in orange and net load in blue: without PVs the base load would be around 1.2 MW, but the PV generation leads to negative net loads of about 0.5 MW.

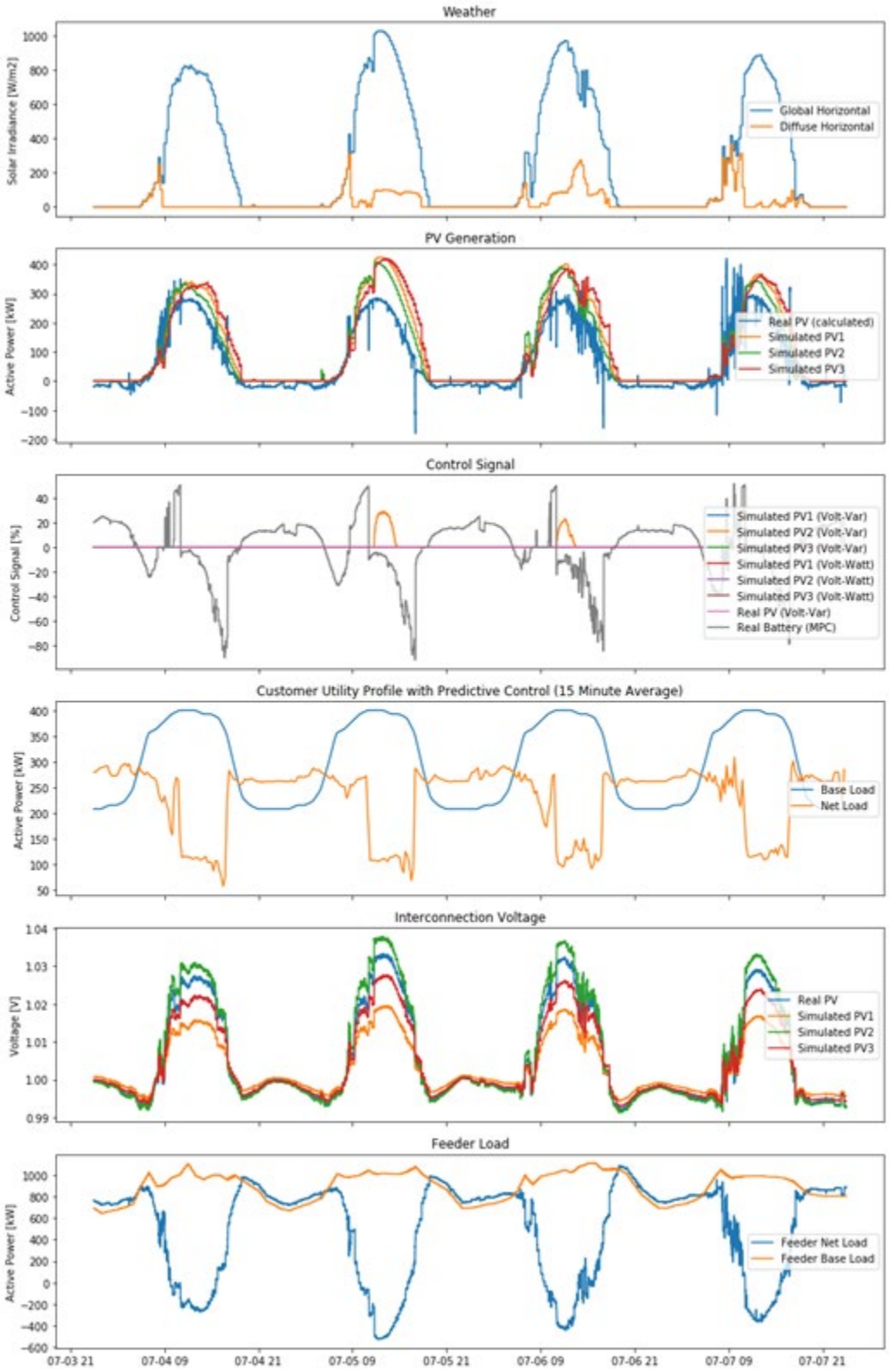

Figure 37: Results from HIL testing involving three simulated PV systems based on real weather data and one real PV system at Flexgrid (controlled with MPC to minimize the total energy cost.

Converting an additional feeder model to ePHASORSIM

The feeder models used in the CyDER project were provided by PG&E in CYME format. Although ePHASORSIM features an automated tool to import CYME models, not all functionalities are supported, and manual operations are necessary to fully convert them. In particular, all unsupported equipment (e.g., PV) should be removed from the model, and invalid configuration should be fixed (e.g. zero-length lines to signify a busbar, improper syntax, etc.). A final manual check should be performed on the output file produced by CYME to ensure the modifications carried over to the final file. All HIL experiments were carried out using the same feeder model so far. A new feeder model was prepared for future testing. This second distribution feeder is larger (1050 nodes) than the one currently in use. Additional preparatory simulations are still being carried out to determine future simulation scenarios, including PV placement.

Task 3.3: CyDER as co-simulation platform

PyFMI and CyDER simulation modes

As mentioned earlier, CyDER uses PyFMI as the co-simulation master algorithm. PyFMI supports the two FMI modes, namely Co-Simulation (CS) FMUs and Model Exchange (ME) FMUs. The main difference is that a CS FMU includes a numerical solver, whereas a ME FMU includes only the model and relies on an external solver for numerical integration. The CS master algorithm enables forward-stepping simulations, while the ME master enables advanced features such as event handling, rollback, and variable time-step integration, which are typically not possible with CS FMUs. ME is also more efficient in solving algebraic loops formed by multiple coupled FMUs.

The left plot of Fig. 38 shows a simple example of the advantage of ME over CS in catching events, where two FMUs are involved. The first FMU is a sinusoidal signal generator and the second one is an on/off controller depending on the value of the signal. As shown in the plot, the ME master is able to accurately detect the transitions in the control signal, whereas the performance of the CS master significantly depends on the (constant) simulation time step. The right plot of Fig. 38 shows the simulation time against the computer clock during a ME simulation. The spikes indicate instances where the master algorithm makes jumps in the integration time step and then rolls back in time, if the time step was found to be too aggressive.

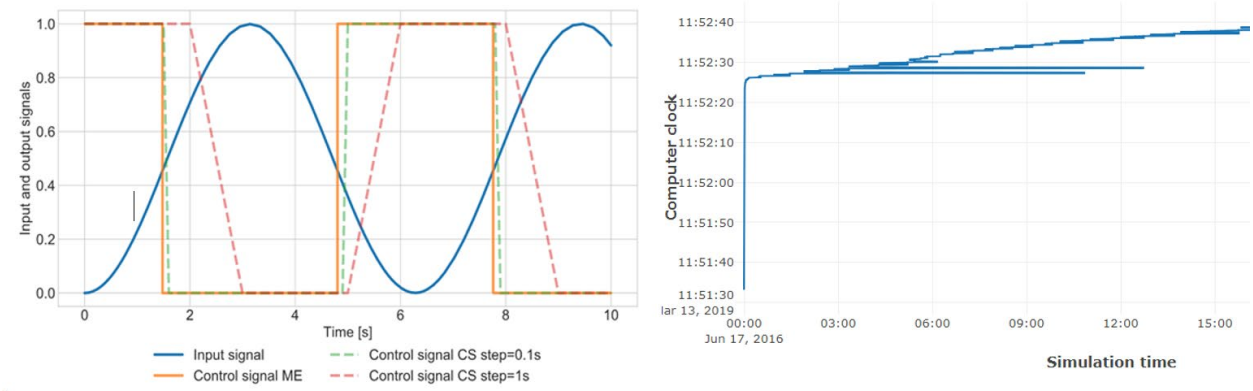

Figure 38: Left: Comparison of a model exchange versus a co-simulation master. Right: example of ME master's ability to roll back and forth in simulation time by using a variable time step.

During BP2 and the first half of BP3, CyDER simulations were run with the CS master, because the ME master was not fully supported. The CS master was able to simulate the PG&E feeder models for modest PV penetration scenarios, for which the algebraic loops from the PV inverters were not stiff due to the small amount of reactive power support and active power curtailment. With very high PV penetration, the CS master sometimes failed to return correct results, as non-decreasing oscillations were observed in the voltages. These oscillations were the result of aggressive Volt/Watt control settings and large amounts of curtailed active PV power.

To resolve this issue, we came up with an approach to sequentially limit the active and reactive power control gains between 2 consecutive iterations (initially 20% of inverter rating, then 10%, 5%, and so on until convergence). Convergence was achieved as soon as the differences of voltage, active power, and reactive power between two consecutive iterations were below a desired threshold. Simulation results with this addition to the CS master are shown in Fig. 39, where the application of gain limiting occurs whenever the magnitude of voltage oscillations decreases, e.g., at the highlighted interval in orange. A challenge that remained with this approach is that achieving voltage convergence might push active and reactive power to diverge from previously converged values, as illustrated at the bottom plot of Fig. 39.

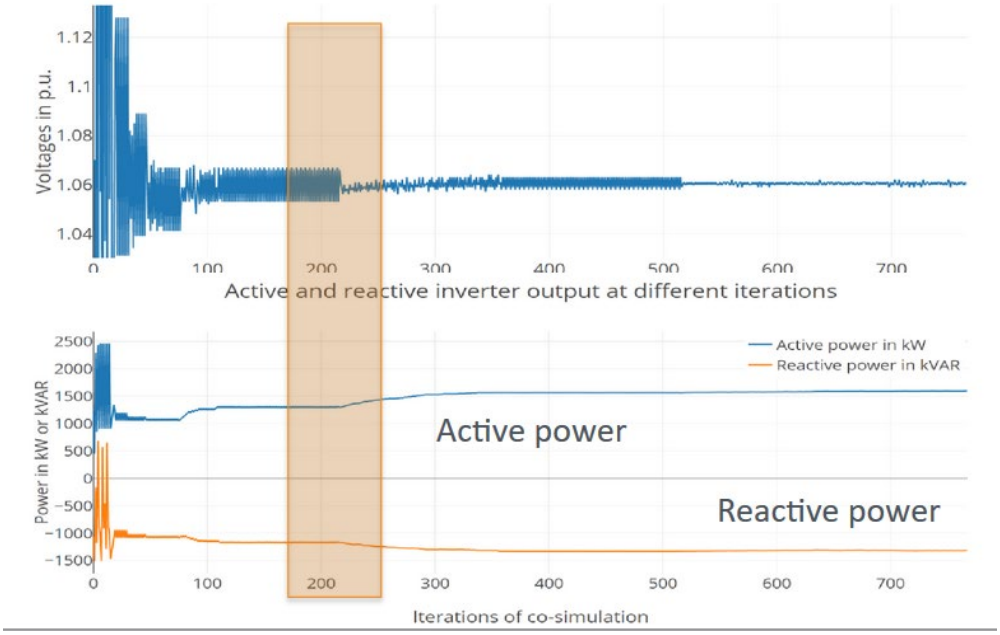

Figure 39: Top: Voltage values during iterations towards convergence for a specific co-simulation time step. Bottom: the resulting inverter active and reactive power at the point of connection.

In the second half of BP3, effort was put on developing a working ME master version to avoid the need of customizing the CS master, as outlined above and also to speed up the simulations. At earlier stages of the project, the FMUs created by SimulatorToFMU had slow simulation times and were often stalling when running in ME mode. In BP3, SimulatorToFMU was improved and as a result these problems were solved and made further development of the ME master possible.

While the CS master runs one FMU after another, the grid FMU and all connected FMUs have their own solvers. These FMUs keep on running one after another, until an

equilibrium point is reached. Because grid and connected FMUs can create feedback loops, e.g. when PV FMUs have controls depending on grid voltages, this can result in oscillations. These feedback loops are difficult to solve in CS mode, especially with high PV penetrations and therefore big changes in power and voltages.

The ME mode solves these problems, because it has a global solver, which can solve existing feedback loops. A connection matrix is generated, which connects all FMUs and the global solver simulates all FMUS simultaneously instead of running one after another with individual solvers. To launch the simulation, the connection matrix is saved as a CSV file and then imported by the ME master. Fig. 40 illustrates a feedback loop of a PV and a grid FMU. This can be represented by the simple connection matrix in Table 1. In every row, the FMU which outputs a variable and the FMU which inputs a variable are specified. This contains the same information than a block diagram with arrows connecting inputs and outputs.

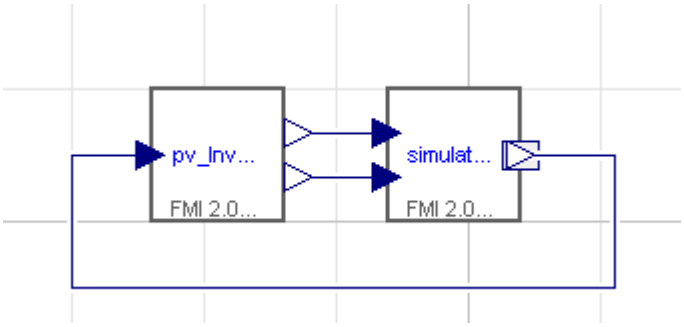

Figure 40: Feedback loop of a PV and a grid FMU

Table 1: Connection matrix for a PV and a grid FMU

FMU1 ID	**FMU2 ID**	**FMU1 Output**	**FMU2 Input**
Grid FMU	PV FMU 1	Voltage at node 10	Voltage
PV FMU 1	Grid FMU	P	P at node 10
PV FMU 1	Grid FMU	Q	Q at node 10

With this ME master, new scenarios which could not be simulated in CS mode, are now possible. One of these scenarios is using battery storage. Because of the small time steps the CS master uses to converge to an equilibrium point, when oscillations in voltages are detected, faulty results were obtained with the CS master when storage is implemented. Another usage for the ME master are scenarios with extreme PV penetration, where the CS mode failed to reach a stable equilibrium point.

Server architecture and user friendliness

We also designed a server architecture for CyDER that allows FMUs to run on multiple computers. This architecture can be seen as a generalization of the previously developed SimulatorToFMU utility: with this architecture, FMUs can connect to a simulator via HTTP requests. This architecture was integrated into SimulatorToFMU, so that the same tool can generate different kind of FMUs.

Moreover, in order to facilitate the use of FMUs we created two utility functions for CyDER. The first utility is called “compile command line” and is helpful to easily generate a new FMU structure. For instance, multiple FMUs can share the same code (CYME FMU, or PandaPower FMU), but sometimes they will represent different distribution grids, therefore inputs and outputs will refer to different nodes, thus changing the FMU structure (inputs/outputs). The compile command lets the user define an Excel file to describe inputs/outputs, thus automating the process of compiling FMUs. The second utility is called “simulate command line” and is helpful to simplify the process of running simulations. It avoids the need to manipulate PyFMI objects and connect FMUs together through a Master. The simulate command line uses the FMU mapping from an Excel file in order to launch a simulation and return all results.

Furthermore, comprehensive documentation with a “getting started” section and examples were written and published online through our GitHub repository and “Read The Docs” https://fmi-for-power-system.readthedocs.io/en/latest/. The documentation includes a section on the vision behind the CyDER project, and the accomplishments made to enable power system simulations with the FMI standard. The documentation then provides a getting started section for installing SimulatorToFMU, JModelica and launching a simple simulation. Code developed throughout the project is available at GitHub: https://github.com/LBNL-ETA/fmi-for-power-system.

New FMUs developed

During BP3 the team’s goal was to develop at least 5 new FMUs to be added to the FMU library available from BP1 and BP2. The new FMUs include: a battery FMU, a micro-PMU data loader, an FMU for blockchain support, and a Pandpower FMU (as an alternative distribution grid simulator besides CYMDIST), the Volt/Var/Watt/Watt inverter FMU (with integrated micro-storage) and, finally, a revised Opal-RT FMU (which allows a Modbus-based communication with the real-time model to work around the limited number of OPAL-RT API calls observed during BP2).

FMI support of power system tools

Within this task, we researched power system simulators that already support the FMI standard and can therefore be readily integrated in CyDER. This list includes the following simulators: EMTP-RV, EcosimPro, Matlab/Simulink, and ESI SimulationX. In addition, we compiled a list of established power systems simulation tools that are distributed with APIs that are readily compatible with CyDER. This list includes the following simulators: CYME, PowerFactory, PSCAD, DSATools, PSS/E, Pandapower, GridDyn, MATPOWER, and OpenDSS.

HELICS-CyDER integration

HELICS is joint effort as part of GMLC to combine the best practices and lessons learned through co-simulation into a single platform shared amongst the National Labs and others. HELICS allows rapid coupling of a wide assortment of different simulations in a flexible manner. Enabling the linkage between HELICS and CyDER (as well as FMI in general) allows tremendous flexibility in designing co-simulations and choosing tools. Integrating HELICS with CyDER allows tools which might not have an FMU wrapper to link in with the CyDER environment and can allow CyDER greater flexibility in dividing up

co-simulations between multiple platforms. This provides significant benefits to both HELICS and CyDER.

The last year of the project included an effort to integrate CyDER and HELICS. Specifically, the HELICS-FMI interface library and tool has been developed by the LLNL team and publicly released at www.github.com/GMLC-TDC/HELICS-FMI. Several FMUs provided by the LBNL team were executed under HELICS-FMI with the results identical to within numerical tolerances to those reported under jModelica (this initial testing included a very simple co-simulation example with a sinusoidal signal generator and an on/off controller depending on the value of the signal). At a later stage, the testing involved power systems related FMUs, as depicted in Fig. 41. The test system includes two FMUs creating an algebraic loop, a PV FMU with Volt/Var/Watt controller, and a distribution grid FMU (exported from pandapower). These FMUs were simulated with HELICS (in co-simulation mode) and the results showed some oscillations with high PV generation, which start when the voltage is high enough to curtail active power, as shown in Fig. 42. The HELICs team reported that the P and Q graph on the left side of Fig. 13 had the same oscillations, but they removed it from the plot.

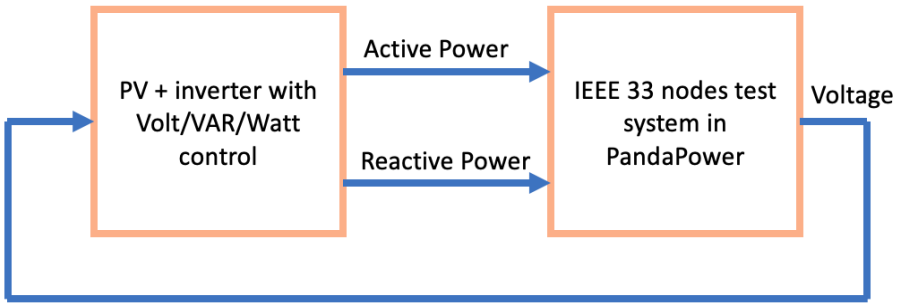

Figure 41: Diagram of the test system.

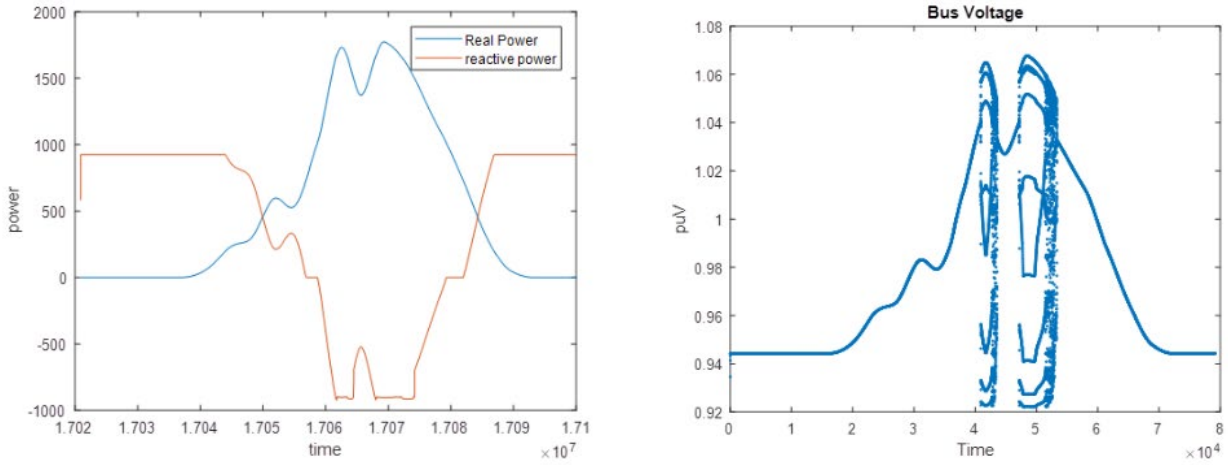

Fig. 42: Simulation results from HELICS: active/reactive power and voltage.

The same simulations were repeated using CyDER (with PyFMI master) both in the Co-Simulation (CS) and Model Exchange (ME) mode. With PyFMI CS we experience the same oscillations around the same time of the day, as in HELICS. The oscillations can be explained considering the dynamic of CS, where FMUs run one after another. In

this case, high voltage results in a curtailment of active power, which leads to decreased voltage values. In turn, these low voltages result in zero curtailment which introduces overvoltages again, and this results to an oscillatory behavior. The oscillations in CS mode cannot be avoided because no global solver exists, which would find a point where these variables could converge to. This shows the big disadvantage of running FMUs in CS mode compared with running them in ME mode. In fact, ME represents the underlying dynamics more realistically, because in reality PVs and grid would run simultaneously and not one after another too. Indeed, the results obtained with ME are shown in Fig. 43 and there are no oscillations.

Note that HELICS does not support the ME yet, so there are no results from the LLNL side to compare ME simulations across CyDER and HELICS. In the final months of the project the code has been updated to work with later releases of HELICS along with updates for easier compilation and execution, more flexible input control, and better integration. Simple documentation is included to allow users a first cut at executing a co-simulation with and FMU in HELICS.

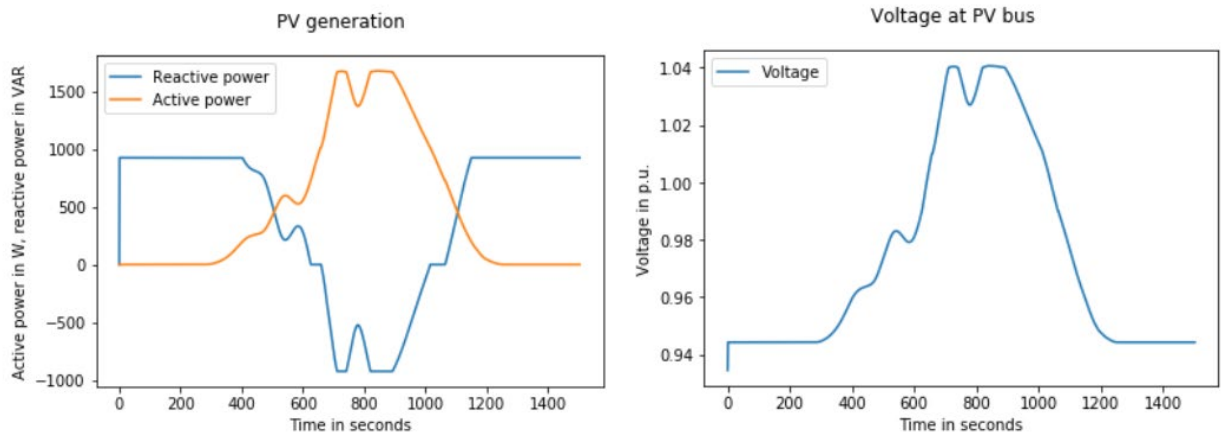

Fig. 43: Left: active power (orange) and reactive power (blue). Right: the voltage at the PV bus. No oscillations are observed in ME co-simulation mode (with CyDER).

Overview of project status at the end of BP3: The relevant milestones M3.1.1, M3.2.1 and M3.3.1 were successfully met. The project passed the Go/no-go decision point, as the milestones were met, a report on the PV penetration potential for 52 feeders in the territory of the partnering utility - with and without controls, voltage results from the closed-loop HIL test are within 5% of the simulated ones, the HIL test ran continuously for more than 24 hours, 5 new FMUs added in CyDER during BP3, and CyDER became available online with its documentation.

5. Significant Accomplishments and Conclusions

The most significant accomplishments and conclusions throughout this project are:

- Development of several power systems related FMUs including distribution and transmission grid models, PV inverters with Volt/Var/Watt controllers, and batteries. In addition, an MPC controller developed within another project was

exported as an FMU, including the associated forecasting function. Overall, this was one of the first projects to investigate the usage of the FMI standard in the power systems area, and the results are promising.

- Development of a co-simulation platform based on PyFMI (together with some visualization functionalities) to analyze the PV hosting capacity of distribution grids without any controls, with smart inverter controls, with EVs, and with battery storage. The platform was used in case studies with 52 PG&E feeders and simulations revealed that Volt/Var control will be sufficient in most cases to resolve overvoltage issues due to the expected increase in PV penetration.
- Further development of SimulatorToFMU, a Python utility which exports Python-driven simulation tools and Python scripts as FMUs. This utility is instrumental in integrating in CyDER third-party tools with Python APIs including exporting OPAL-RT, a real-time grid simulation tool, and enhances future usability of CyDER after project completion.
- Development of supporting code (cyder-s and cyder-c utilities) to automate and facilitate setting up and executing co-simulations with CyDER, especially in the Model Exchange mode (which was found to perform better than the co-simulation mode during the analyses of this project).
- FMI-based coupling of software simulators (e.g., PV and inverter models) and hardware components (OPAL-RT grid simulator and Ametek grid emulator) in HIL co-simulations, and delivering an experimental setup that can be used by other researchers. This is a significant contribution to the FLEXGRID facility at LBNL.
- Initial integration effort between the HELICS and CyDER platforms. FMUs developed within CyDER were integrated in HELICS and simulation results across the two platforms were identical.

6. Inventions, Patents, Publications, and Other Results

The following publications were made during the course of this project:

- Journal paper 1: T. S. Nouidui, J. Coignard, C. Gehbauer, M. Wetter, J. Y. Joo, E. Vrettos. CyDER – an FMI-based co-simulation platform for distributed energy resources, Journal of Building Performance Simulation, September 2019, 12(5):566-79.
- Journal paper 2: J. Coignard, P. MacDougall, F. Stadtmueller, E. Vrettos. Will Electric Vehicles Drive Distribution Grid Upgrades?: The Case of California, IEEE Electrification Magazine, June 2019, 7(2):46-56.
- Conference paper 1: J. Coignard, T. Nouidui, C. Gehbauer, M. Wetter, J-Y. Joo, P. Top, RR. Soto, B. Kelley, E. Stewart. CyDER - A Co-Simulation Platform for Grid Analysis and Planning for High Penetration of Distributed Energy Resources, IEEE PES Power & Energy Society General Meeting, August 2018.
- Conference paper 2: T. Nouidui, and M. Wetter. SimulatorToFMU: A Python Utility to Support Building Simulation Tool Interoperability, Proceedings of the 2018 Building Performance Analysis Conference and SimBuild, Chicago, IL.
- Conference paper 3: E. Vrettos and C. Gehbauer. A Hybrid approach for short-term PV power forecasting in predictive control applications, IEEE Powertech conference, Milan, Italy, June 2019.

In addition, another journal paper focusing on the MPC controller integration in CyDER is currently in preparation. Within the course of the project, the following conference and workshop presentations on the CyDER platform and its applications were also made, but without publishing a paper:

- T. Nouidui, NA-MUG Conference on System Modeling and Simulation with Open Standards, Stanford University, 2017.
- J. Y. Joo, DOE SunShot Workshop: Numerical Analysis Algorithms for Distribution Networks, Argonne National Laboratory, 2017.
- J. Coignard and F. Stadtmueller, i-PCGRID Conference, Session: Sustainable and Distributed Resources and Green Power, PG&E Headquarters, SF, 2018.
- E. Vrettos, Workshop on Data Driven Analytics, SLAC, Menlo Park, CA, 2018.
- E. Vrettos, Workshop on Challenges for Distribution Planning, Operational and Real-time Planning Analytics, DOE, Washington DC, 2019.

Finally, the work on EV impact analysis on distribution grids initiated a new collaboration with NRDC.

7. Future Opportunities for Research

There are multiple avenues for future research and development in the co-simulation field, including additional use cases for CyDER. The primary objective of CyDER has been enhancing user friendliness and obtaining stable co-simulation performance even in the presence of aggressive inverter controls. Nevertheless, scalability has been a secondary aspect in this project and this could well be the main focus in follow-up work. One example is making necessary adaptations to CyDER and then using it in a supercomputing setting to perform large-scale co-simulations involving a large number of distribution feeders (with active Volt/Var/Watt controllers) connected to the same substation.

Another interesting application is using CyDER to simulate transmission-scale low-inertia inverter-dominated grids. In such a scenario, different modeling approaches might be necessary for different parts of the grid, depending on the inverter penetration level and their effect on the overall power system dynamics. More precisely, parts of the grid with low inverter shares could be modeled with traditional phasor-based models, whereas more detailed Electro-Magnetic Transient (EMT) models might be required for parts of the grid with massive inverter penetration. Since the time steps used in EMT modes are orders of magnitude smaller than those in phasor-based models, such a scenario provides a good use case for application of co-simulation techniques.

Besides simulation, it would be interesting to investigate CyDER capabilities for optimization applications. A first step in this direction was taken during the project by exporting an optimization-based MPC controller as FMU and integrating it in CyDER. Even though this was a rather ad-hoc approach, it would be interesting to encapsulate generic optimization problems with well-defined input-output relationship as FMUs, so that they can be routinely used as part of larger simulation programs, e.g., to model optimal controllers. This can include both mathematical (derivative-based) optimization and heuristic optimization techniques. Potential applications include optimal placement of

a finite amount of micro-PMU sensors along a feeder, optimal placement and setpoints of DERs, or other supervisory predictive control decisions.

Last but not least, future work could intensify the integration effort between CyDER and HELICS. So far, HELICS and CyDER have been shown to produce identical results when simulating FMUs in the CS mode. However, multiple analyses within the CyDER project identified the limitations of CS mode and highlighted the advantages of the ME mode. Therefore, future work could focus on developing ME support within HELICS and verify the validity of implementation using CyDER as a benchmark. This can be a future project to harmonize the research activities in the co-simulation area.

8. References

1. N. Hadjsaid, and J-C. Sabonnadiere, "Smart Power Grids", 2012, ISBN 978-1-84821-261-9.
2. S. Chatzivasileiadis, et al., "Cyber–Physical Modeling of Distributed Resources for Distribution System Operations", 2016, Proceedings of the IEEE 104 (4): 789–806. doi:10.1109/JPROC.2016.2520738.
3. C. Gomes, C. Thule, D. Broman, P. Gorm Larsen, and H. Vangheluwe, "Co-simulation: State of the Art", 2017, https://arxiv.org/abs/1702.00686.
4. J. Daily, S. Ciraci, J. Fuller, and K. Agarwal, "FNCS: A Framework for Power System and Communication Networks Co-Simulation", Proceedings of the Symposium on Theory of Modeling & Simulation - DEVS, Integrative, Tampa, Florida, 36:1–36:8, 2014.
5. Sandia National Laboratories, "SCEPTRE", 2016, https://www.osti.gov/servlets/purl/1376989.
6. B. Zeigler, and S. Vahie, "DEVS Formalism and Methodology: Unity of Conception/Diversity of Application." Proceedings of the 25th conference on Winter simulation, Los Angeles, CA, 1993.
7. M. Saurabh, R. Mark, P. Annabelle, L. Monte, K. Dheepak, and J. Wesley, "A System-of-Systems Approach for Integrated Energy Systems Modeling and Simulation." Proceedings of the Conference on Summer Computer Simulation, Chicago, IL, 2015.
8. ZeroMQ, 2015, http://zeromq.org.
9. B. Palmintier, Bryan, E. Hale, T. M. Hansen, W. B. Jones, D. Biagioni, H. Sorensen, H. Wu, and B-M. Hodge, "IGMS: An Integrated ISO-to-Appliance Scale Grid Modeling System", IEEE Transactions on Smart Grid 8 (3): 1525–1534, 2017, doi:10.1109/TSG.2016.2604239.
10. M. Büscher, et al., "Integrated Smart Grid Simulations for Generic Automation Architectures with RT-LAB and Mosaik", 2014 IEEE International Conference on Smart Grid Communications (SmartGridComm), Venice, 194–199, doi:10.1109/SmartGridComm.2014.7007645.
11. R. Roche, S. Natarajan, A. Bhattacharyya, and S. Suryanarayanan, "A Framework for Co-simulation of AI Tools with Power Systems Analysis Software", 23rd International Workshop on Database and Expert Systems Applications, Vienna, 350–354, 2012, doi:10.1109/DEXA.2012.9.

12. T. Godfrey, M. Sara, R. C. Dugan, C. Rodine, D. W. Griffith, and N. T. Golmie, “Modeling Smart Grid Applications with Co-Simulation”, First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD, 291–296, 2010, doi:10.1109/SMARTGRID.2010.5622057.
13. J. G. Gomez-Gualdron, and M. Velez-Reyes, “Simulating a Multi-Agent based Self-Reconfigurable Electric Power Distribution System”, IEEE Workshop on Computers in Power Electronics, Troy, NY, 1–7, 2006, doi:10.1109/COMPEL.2006.305644.
14. T. Blochwitz et al, “Functional Mockup Interface 2.0: The Standard for Tool Independent Exchange of Simulation Models”, Proceedings of the 9th International Modelica Conference, Munich, 173–184, 2012.
15. C. Andersson, J. Åkesson, and C. Führer, “PyFMI: A Python Package for Simulation of Coupled Dynamic Models with the Functional Mock-up Interface”, Technical Report in Mathematical Sciences; Vol. 2016, No. 2. Centre for Mathematical Sciences, Lund University, 2016, https://pypi.org/project/PyFMI/.
16. T. Nouidui, and M. Wetter, “SimulatorToFMU: A Python Utility to Support Building Simulation Tool Interoperability”, Proceedings of the 2018 Building Performance Analysis Conference and SimBuild, Chicago, IL, 2018, https://github.com/LBNLETA/SimulatorToFMU.
17. Ptolemaeus, C., Editor, System Design, Modeling, and Simulation Using Ptolemy II, Ptolemy.org, 2014.
18. J. Coignard, T. Nouidui, C. Gehbauer, M. Wetter, J-Y. Joo, P. Top, RR. Soto, B. Kelley, E. Stewart, CyDER-A Co-Simulation Platform for Grid Analysis and Planning for High Penetration of Distributed Energy Resources, IEEE PES Power & Energy Society General Meeting, August 2018.
19. E. Vrettos and C. Gehbauer, A Hybrid approach for short-term PV power forecasting in predictive control applications, IEEE Powertech conference, Milan, Italy, June 2019.
20. J. Coignard, P. MacDougall, F. Stadtmueller, E. Vrettos, Will Electric Vehicles Drive Distribution Grid Upgrades?: The Case of California, IEEE Electrification Magazine, June 2019, 7(2):46-56.
21. T. S. Nouidui, J. Coignard, C. Gehbauer, M. Wetter, J. Y. Joo, E. Vrettos, CyDER – an FMI-based co-simulation platform for distributed energy resources, Journal of Building Performance Simulation, September 2019, 12(5):566-79.