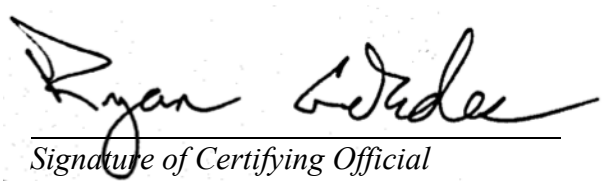
Final Technical Report (FTR)

Cover Page

a. *Federal Agency*	Department of Energy	
b. *Award Number*	DE-EE0009338	
c. *Project Title*	Achieving Cyber-Resilience for Power Systems using a Learning, Model-Assisted Blockchain Framework	
d. *Recipient Organization*	Virginia Polytechnic Institute and State University	
e. *Project Period*	*Start: 05/01/2021*	*End:* 04/30/2023
f. *Principal Investigator (PI)*	Ryan M. Gerdes Associate Professor rgerdes@vt.edu 571-858-3106	
g. *Business Contact (BC)*	Orlando Florez Program Manager oflorez@vt.edu 571-858-3336	
h. *Certifying Official (if different from the PI or BC)*	N/A	



Signature of Certifying Official *2024-01-18*

Date

By signing this report, I certify to the best of my knowledge and belief that the report is true, complete, and accurate. I am aware that any false, fictitious, or fraudulent information, misrepresentations, half-truths, or the omission of any material fact, may subject me to criminal, civil or administrative penalties for fraud, false statements, false claims or otherwise. (U.S. Code Title 18, Section 1001, Section 287 and Title 31, Sections 3729-3730). I further understand and agree that the information contained in this report are material to Federal agency's funding decisions and I have any ongoing responsibility to promptly update the report within the time frames stated in the terms and conditions of the above referenced Award, to ensure that my responses remain accurate and complete.

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 ENERGY TECHNOLOGIES OFFICE FISCAL YEAR 2020 FUNDING PROGRAM Award Number(s) DE-EE0009338."

Executive Summary

The secure integration and management of distributed energy resources (DER) and power aggregators in the electric grid requires secure communications and a physics-aware Command and Control (C2) strategy. A Blockchain (BC)-based overlay network was developed to provide a security layer for the existing power grid network that mitigates risks in current and legacy network and C2 protocols. By integrating a Model-Assisted Machine Learning (MAML) framework with a Secure Blockchain Overlay Network (SBON) a defense-in-depth strategy was achieved.

In our approach, the MAML framework leveraged a smart contract framework to gather network data and learn the dynamics of DER to develop detection strategies for attacks targeting sensors and actuators used by DER.

The MAML framework learned dynamical systems models for individual DERs to detect sensor attacks. For DER we utilized a Digital Twin (DT) to accelerate the learning process for a model resistant to stealthy attacks. The project created DT for PV inverters and BESS. The DTs were coupled with a model-assisted, data-driven learning of DER behavior. Specifically, we evaluated architectures for model-based learning with model-free fine-tuning. Additionally, differential privacy techniques were used to obfuscate data, while still allowing the computation of attack detection results based on obfuscated data.

The SBON developed leverages a private permissioned blockchain network orchestrated with the Hyperledger Fabric framework. To connect the cyber world, which orchestrates the blockchain fabric, and the physical world where the power network resides, we developed a system implementation to enable the secure interaction of the physical world and the abstracted blockchain.

Table of Contents

1. Table of Contents

Table of Contents **5**

2. Background **6**

3. Project Objectives **7**

4. Project Results and Discussion **9**

5. Significant Accomplishments and Conclusions: **41**

6. Path Forward **41**

7. Products **42**

8. Project Team and Roles **42**

9. References **44**

2. Background

Model-based attack detection for cyber-physical systems (CPS) [3] has been used to protect numerous energy-related systems, e.g., [4]. Model-based approaches, however, are limited by modeling uncertainty (i.e., unmodeled dynamics) and measurement noise, both of which an attacker can exploit to remain undetected [1]. Recently, learning-based approaches have been proposed to minimize modeling uncertainty [5] and have found some success in applications related to power regulation [6] and security [7]. Learning-based approaches, however, can be slow to converge; require substantial interactions (measurements) with the environment; may not observe relevant dynamics during offline training; during online learning are vulnerable to attacker poisoning; or may introduce system brittleness as an artifact of the learning process [8]. To address these limitations, we explored a model-assisted, learning-based [9] attack detection approach. While maintaining the capabilities of a traditional model-based approaches, our approach can identify evasive attackers, and localize impacted DER more quickly than traditional model-based or existing learning approaches.

Deployment of a Secure Blockchain Overlay Network (SBON) as an overlay network on the existing distribution network has the potential to address long-standing challenges of poor/absent authentication, encryption, and identity management for power systems. To date, blockchain (BC) has been applied for transactive energy exchange, but not as an interoperable and unified control framework by utilities or grid operators [10]. A blockchain-based approach has advantages over other industry initiatives that attempt to provide these benefits, such as Open Field Message Bus [11], due to greater security provided by nearly a decade of secure reference implementation development and code auditing of blockchain protocols. The conventional 51% attacks on Blockchain networks [12] do not apply to the proposed BC/SBON architecture due to the use of Practical Byzantine Fault Tolerance, a permissioned consensus algorithm that relies on cryptographic identity to verify transactions [13]. By implementing the SBON in a trusted hardware module we can also achieve secure off chain execution.

While secure multiparty computation (SMC) techniques have been explored for other applications (e.g., financial trading), the aggregator-level attack detection problem presents unique challenges. The large amount of data and the non-linear operations makes it inefficient to use SMC. On the other hand, existing differential-privacy based data release methods are applicable to answering statistical queries, but here

the accuracy of the attack detection depends on the accurate release of individual DER's data records, which is at odds with privacy. Our proposed privacy-preserving synthetic data release algorithms achieved high accuracy while preserving the privacy of individual users by leveraging the unique characteristics of the grid network topology graph to anonymize the data exchanged with and between aggregators.

While the use of blockchain in energy is relatively well studied, much of the existing work focuses on supporting energy trading [14], and little consideration has been given on leveraging it for resiliency in power network. Besides scalability, the seemingly conflicting properties between the closed on-chain system of blockchain in the cyber world and the need to interact with the physical world in CPS resiliency is also a challenge. To bridge this gap, we build on top of existing work [2] both in theoretical construction and system implementation to enable composition of on-chain and off-chain functions.

3. Project Objectives

Two key technologies to achieve cyber-resilience for power systems were developed: a BC-enabled secure communication and management architecture and a MAML framework for attack detection and response. These two security enablers could be deployed to DER and leveraged at the distribution/transmission grid level to ensure grid network resiliency (protect, detect, and respond) against sophisticated-tier threats involving compromised DER and aggregators.

The threat model considered assumed DER compromise at the local device level (sensors, device firmware), plus global aggregator level (DER controllers, actuators, DER management systems (DERMS)), and utility level (compromised aggregators, DERMS, subregions, and microgrids). Local and global attacks were considered, including false data injection, spoofing, and privilege escalation to achieve one or more of the following cyber-physical effects: (1) sub-synchronous resonance, (2) amplification of weak grid conditions, (3) load shedding, and (4) inter-area oscillations.

At the level of individual DER the MAML could be used to detect sensor attacks and determine the proper response to potentially malicious C2 signals (i.e., whether an actuator attack is underway that would negatively impact the grid). As individual DER may not be able to detect all sensor/actuator attacks, and DER themselves

may be compromised, the MAML detection framework would also be used by aggregators/utility operators. As these actors have more computational resources and have access to sensor data from multiple DER (i.e., a system view), they have the ability to execute more complex algorithms that leverage a larger set of data to determine which DER are compromised and how to respond (e.g., isolate affected DER or island distribution networks) without jeopardizing grid operations.

In our MAML framework, attack detection at both local and aggregator levels assumed the collection and sharing of electric usage data from individual DERs and aggregators. However, this raised privacy concerns. A DER may not want to let nearby DERs know about its electric usage, and an aggregator may not want to share all its data with another aggregator since they may belong to mutually untrusted companies/organizations. We proved techniques to enable joint detection among untrusted parties without compromising the privacy of users.

Several components of a Secure Blockchain Overlay Network (SBON) were developed to address cybersecurity vulnerabilities arising from vulnerable, heterogeneous, and non-interoperable command-and-control (C2) protocols used by DER, aggregators, and utility and operator distribution systems. The SBON allows for secure configuration of a peer-to-peer ledger protocol with smart contract support (e.g., a private permissioned blockchain using the Ethereum blockchain protocol).

3.1 Task Summaries

3.1.1 Task 1.0: Digital Twin Development

Digital twin (DT) required for model-based detection to be built. The Twins can be used at the edge or at the Utility level. DT will be transferred to prime in an executable format with the appropriate interfaces to receive inputs and publish estimates.

3.1.2 Task 2.0: Learning Framework for Sensor Attack Detection

Designing and developing the MAML framework, including integration of DT, to enable detection of attacks against DER sensors by comparing agent-predicted behavior to observed behavior. Initial experimental validation of approach conducted on existing inverter and BESS designed by partner and with final validation performed on utility-grade DER.

3.1.3 Task 3.0: Grid Modeling

Reference load, voltage and frequency profiles of the utility partner(s)' service-area grid to be determined.

3.1.4 Task 4.0: BC Framework (BCF) Development

R&D of a private permissioned peer-to-peer ledger and smart contract framework ('BCF'). The BCF to provide protection (secure communications, identity management, secure updates, network segmentation), and enable automated detection and response using a smart contract framework. BCF will act as an overlay to the existing power grid network (e.g., DNP3/IP and SCADA) to provide a unified and interoperable secure architecture for utility integration and management of aggregators and DER.

3.1.5 Task 5.0: Enabling Physical World Access from Blockchain

New theory and implementation to enable smart contracts to execute off-chain functions in individual DER, using the notion of local consensus, in which smart contract functions are off-loaded to nodes with trusted modules who can attest to interested parties the trusted execution of commands.

3.2 Go/No-Go Decision Points and Milestones

PV Inverter Digital Twin model accuracy; BESS Digital Twin model accuracy; PV Inverter MAML accuracy; BESS MAML accuracy; Transmission network model(s) accuracy; Distribution network model(s) accuracy; Local (DER) level attack detection; Global (aggregator or utility) level attack detection.

4. Project Results and Discussion

4.1 Digital Twin Development (Task 1)

4.1.1 Develop Digital Twin (DT) for PV Inverter (Subtask 1.1)

Outcome: DT that describes the behavior of a PV inverter in the field. A General Electric LV5 inverter (1,500 VDC, 4 MW input capacity), connected to a 115 kV transmission network via a substation, served as the DT target.

4.1.1.1 PV System Description

The PV inverter system assumed in Figure 1 was composed of solar PV panels, dc/dc conversion with Maximum Power Point Tracking (MPPT) capability, inverter, grid connection and plant controller.

Figure 1. PV system

The PV system captured the following scenarios:

- Provide as much power as possible based on PV capacity under specific irradiance (MPPT function).
- Provide fixed power to the grid when power command is lower than the PV capacity (Curtailment/Reserve function).
- Provide commanded reactive power to the grid.

Measured variables and their locations are shown in the figure Figure 2 and Table 1 below.

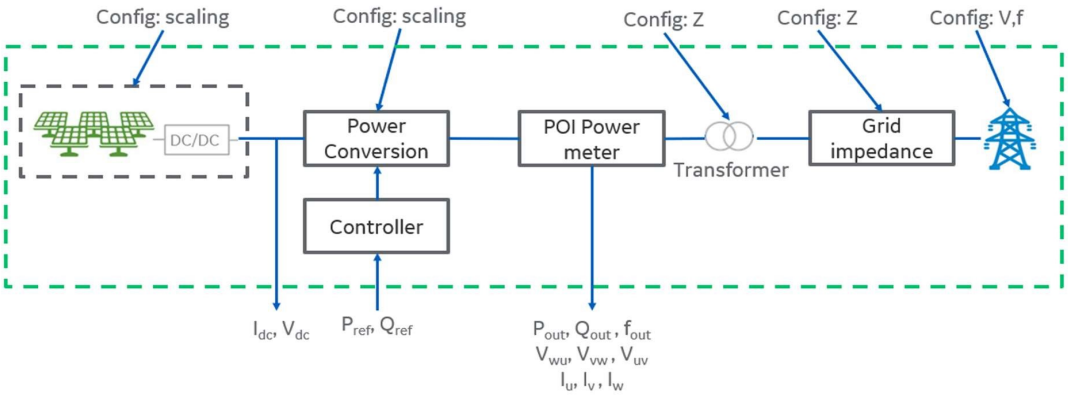


Figure 2. PVI system measurements

Table 1. Measurement signals in PV system

Measured variables	Description
V_{dc} [V]	Measured dc bus voltage
I_{dc} [A]	Measured dc bus current
Pref [kW]	Reference power
Pout [kW]	Inverter output active power
f_{out} [Hz]	System frequency
Q_{ref} [kVar]	Reference reactive power
Q_{out} [kVar]	Inverter output reactive power
V_{wu} [V]	Line-to-line WU voltage
V_{vw} [V]	Line-to-line VW voltage
V_{uv} [V]	Line-to-line UV voltage
I_u [A]	Phase U current

I_v [A]	Phase V current
I_w [A]	Phase W current

4.1.1.2 PV DT Model Results

An experimentally verified PV DT model under Python platform has been delivered. The model can capture both MPPT and curtailment modes.

Table 2 shows the accuracy of the PV DT model in Python platform compared to Matlab under curtailment mode.

Table 2. PV DT Model Accuracy Comparison in Python and Matlab Platforms under Curtailment Mode

Vdc	0.00094%	Idc	-0.1019%	Pout	-0.075%
fout	0%	Vuv	-0.00005%	Iu	-0.1113%

Table 3 shows the accuracy of the PV DT model in Python platform compared to Matlab under MPPT mode.

Table 3. Model Accuracy Comparison in Python and Matlab Platforms under MPPT Model

Vdc	-0.0027%	Idc	0.3756%	Pout	0.3518%
fout	0%	Vuv	-0.00002%	Iu	0.2835%

4.1.1.3 PV Data Summary

The PV data consists of 10 datasets as described in Table 4. The PV system is tested in both MPPT and curtailment/reserve modes with both the steady state and transient behavior recorded. Cases are selected in the report to demonstrate the capability of the PV system.

Table 4. PV Dataset List

Test name	Category	Scenarios	Sampling time (ms)	Time duration (s)
M_Curtail_P_100 kW	PV Reserve	Morning. PV curtailment with output power at 100 kW.	10	25
M_Curtail_P_50 kW	PV Reserve	Morning. PV curtailment with output power at 50 kW.	10	30
A_Curtail_P_100 kW	PV Reserve	Afternoon. PV curtailment with output power at 100 kW.	10	35

A_Curtail_P_50 kW	PV Reserve	Afternoon. PV curtailment with output power at 50 kW.	10	20
EA_MPPT	MPPT	MPPT. Early Afternoon	10	25
LA_MPPT	MPPT	MPPT. Late Afternoon	10	45
M_Curtail_P_100_50 kW	PV reserve step change	Morning. PV curtailment. Transient behavior with Pref from 100 to 50kW.	10	15
A_Curtail_P_100_50 kW	PV reserve step change	Afternoon. PV curtailment. Transient behavior with Pref from 100 to 50kW.	10	15
EA_MPPT_ Curtail w. P_100 kW	MPPT to reserve	Early Afternoon. MPPT to curtailment with output power at 100kW.	10	22
LA_MPPT_Curtail w. P_100 kW	MPPT to reserve	Late Afternoon. MPPT to curtailment with output power at 50kW.	10	25

Test result: PV operates at MPPT mode

The MPPT data for the panels operating in the early afternoon is shown in Figure 3. Bus voltage, current, and power during MPPT. In this case, the system is operating to generate the maximum available power for the operating conditions.

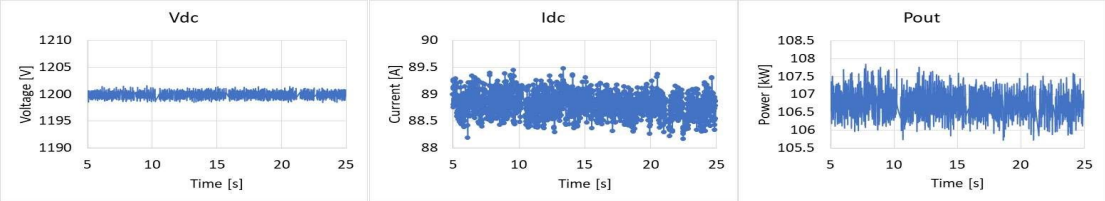


Figure 3. Bus voltage, current, and power during MPPT

Test Result: Mode transition from MPPT to curtailment

The transition from MPPT to curtailment is shown in Figure 4. Bus voltage, current, and power during mode transition. Curtailment with Pref = 100 kW. The test is done in early afternoon. Before the mode transition event, the system is operating to generate the maximum available power for the operating conditions.

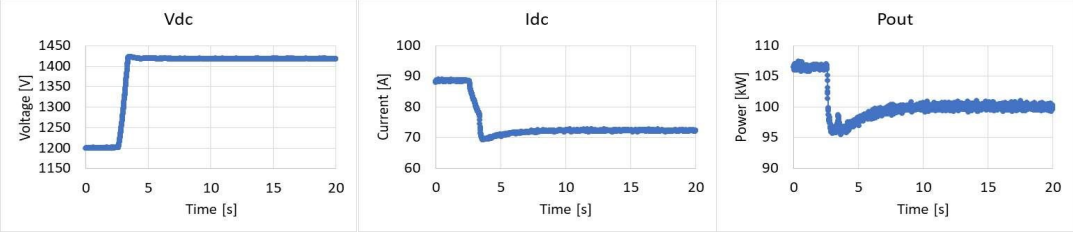


Figure 4. Bus voltage, current, and power during mode transition. Curtailment with Pref = 100 kW

Figure 5 shows another mode transition test done in late afternoon.

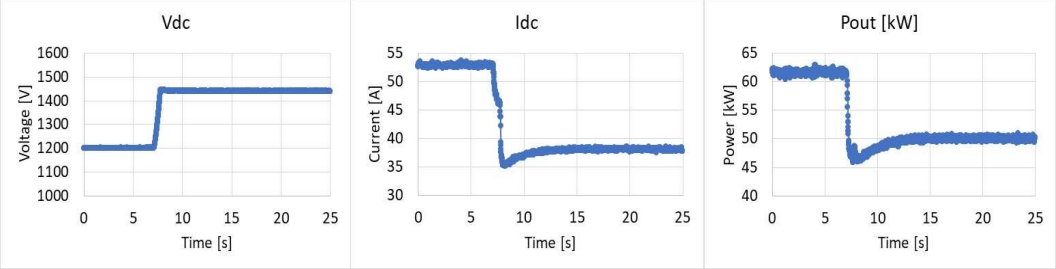


Figure 5. Bus voltage, current, and power during mode transition. Curtailment with Pref = 50 kW

4.1.2 Develop DT for BESS (Subtask 1.2)

Outcome: DT that describes the behavior of a BESS inverter in the field. A General Electric Renewable Energy Reservoir (4 MWh, lithium ion), connected to a 115 kV transmission network via a substation, served as the DT target.

4.1.2.1 BESS System Description

The BESS system is composed of battery racks, dc/dc conversion, an inverter, grid connection and plant controller, as depicted in Figure 6.

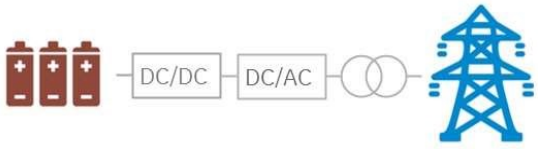


Figure 6. The BESS System

The BESS system captures the following scenarios:

- Discharge the battery to export power to the grid.
- Charge the battery by power injected from the grid.
- Deliver commanded reactive power to the grid.

Measured variables and their locations are shown in Table 5 and Figure 7.

Table 5. Measurement signals in BESS system

Measured Variables [unit]	Description
SOC [%]	State of charge
Status_bat [-]	Battery status
V_{bat} [V]	Measured battery voltage
I_{bat} [A]	Measured battery current
P_{ref} [kW]	Reference power

P_{out} [kW]	Inverter output active power
f_{out} [Hz]	System frequency
Q_{ref} [kVar]	Reference reactive power
Q_{out} [kVar]	Inverter output reactive power
V_{wu} [V]	Line-to-line WU voltage
V_{vw} [V]	Line-to-line VW voltage
V_{uv} [V]	Line-to-line UV voltage
I_u [A]	Phase U current
I_v [A]	Phase V current
I_w [A]	Phase W current

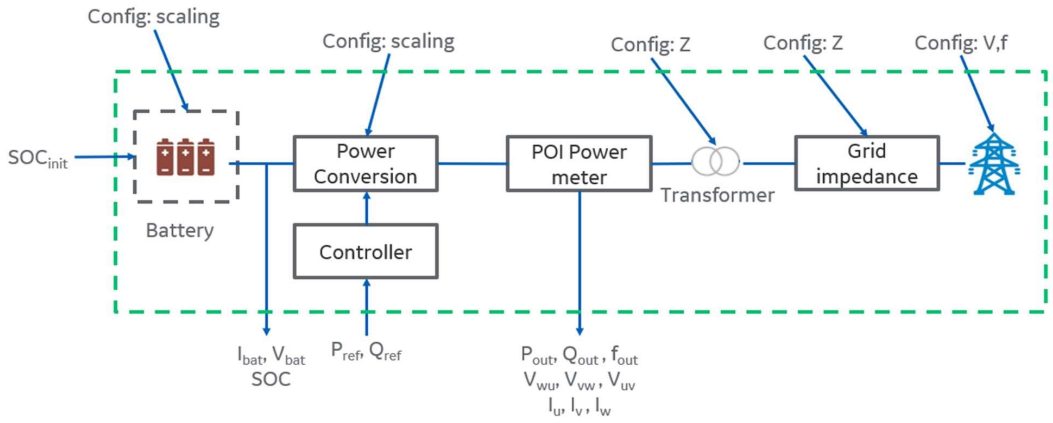


Figure 7. BESS System Measurements

4.1.2.2 BESS DT Model Results

Both steady state and transient behaviors and charging and discharging operations were simulated in the BESS DT model and compared against test data.

Steady State Test Comparison

Comparisons between the test data and DT model are provided in the following tables. As depicted in the table, errors of all simulation output parameters are less than 3%, besides the battery current.

Table 6. Test and model comparison for case with +600 kW active power and 0 kVAR reactive power

Inputs			
Reference Power [kW]	+600	+600	N/A
Pout [kW]	+600	+600	0%
SOC [%]	34.125	34.125	0.00%

Status_bat [-]	-1	-1	0.00%
Ibat [A]	620	635.62	-2.52%
Vbat [V]	958.2	952.8	0.56%
fout [Hz]	59.998	60.06	-0.10%
Vuv [V]	491.2	492.15	-0.19%
Vvw [V]	491.2	492.15	-0.19%
Vwu [V]	491.2	492.15	-0.19%
Iu [A]	705.45	702.25	0.45%
Iv [A]	705.45	702.25	0.45%
Iw [A]	705.45	702.25	0.45%

Table 7. Test and model comparison for case with -600 kW active power and 0 kVAR reactive power

Inputs			
Reference Power [kW]	-600	-600	N/A
Pout [kW]	-600	-600	0.00%
SOC [%]	36.075	36.075	0.00%
Status_bat [-]	1	1	0.00%
Ibat [A]	-587.5	-603.1	2.66%
Vbat [V]	979.6	963.1	1.68%
fout [Hz]	60.002	60.025	-0.04%
Vuv [V]	487.0	488.41	-0.29%
Vvw [V]	487.0	488.41	-0.29%
Vwu [V]	487.0	488.41	-0.29%
Iu [A]	708.95	708.15	0.11%
Iv [A]	708.95	708.15	0.11%
Iw [A]	708.95	708.15	0.11%

Transient Test Comparison

To ensure the DT model accurately captured the dynamic behavior of the physical BESS, model outputs were compared to test data. The normalized root mean squared error was used for model validation.

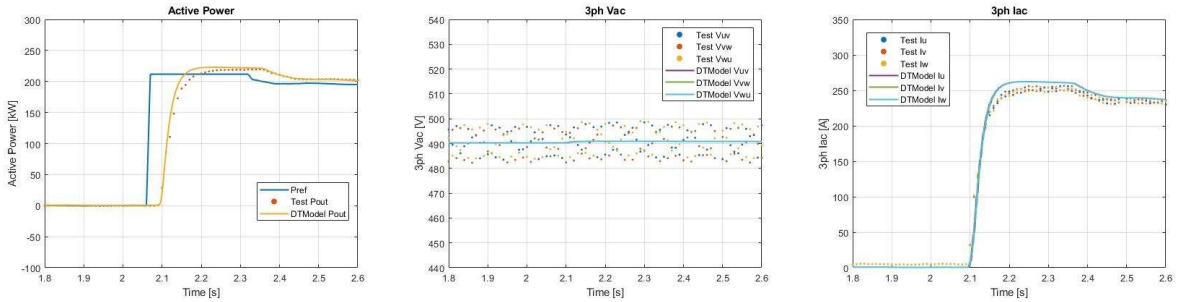


Figure 8. Test and DT model result comparison for case with 0 kW to 200 kW active power step

4.1.2.3 BESS Data Summary

The BESS data consists of 3 main categories with total of 27 datasets as described in Table 8, Table 9, and Table 10. The BESS system captured the following scenarios:

- Active power step tests, with different combinations of initial and target power reference. Reactive power reference is set to be 0 kVar for all cases.
- VAR step tests, with different combinations of initial and target reactive power reference, hold and set back to the initial. Power reference is set to be 0 kW or 400 kW.
- Volts tests, with 100% initial grid voltage to a lower target grid voltage, hold and set back to the initial. Power reference is set to be 0 kW or 400 kW.

Three cases were selected to demonstrate the capability of the BESS system; specifically, battery charging, battery discharging, and reactive power step test, as detailed below.

Table 8. BESS Dataset List - Active Power Step Test

Test name	Scenarios	Sampling Time (ms)	Time Duration (s)
P_+7_+224kW	Qref = 0 kVar. Set Pref from 7 kW to 224 kW	10	40
P_+224_+441kW	Qref = 0 kVar. Set Pref from 224 kW to 441 kW	10	40
P_+441_+657kW	Qref = 0 kVar. Set Pref from 441 kW to 657 kW	10	40
P_+658_+855kW	Qref = 0 kVar. Set Pref from 658 kW to 855 kW	10	40
P_-858_-642kW	Qref = 0 kVar. Set Pref from -858 kW to -642 kW	10	40
P_-641_-426kW	Qref = 0 kVar. Set Pref from -641 kW to -426 kW	10	40
P_-426_-210kW	Qref = 0 kVar. Set Pref from -426 kW to -210 kW	10	40
P_+200_+400kW	Qref = 0 kVar. Set Pref from 200 kW to 400 kW	10	40
P_+400_+600kW	Qref = 0 kVar. Set Pref from 400 kW to 600 kW	10	40
P_+600_+800kW	Qref = 0 kVar. Set Pref from 600 kW to 800 kW	10	40
P_+800_-800kW	Qref = 0 kVar. Set Pref from 800 kW to -800 kW	10	40
P_-800_-600kW	Qref = 0 kVar. Set Pref from -800 kW to -600 kW	10	40
P_-600_-400kW	Qref = 0 kVar. Set Pref from -600 kW to -400 kW	10	40
P_-400_-200kW	Qref = 0 kVar. Set Pref from -400 kW to -200 kW	10	40

Table 9. BESS Dataset List - VAR Step Test

Test name	Scenarios	Sampling Time (ms)	Time Duration (s)
VAR_0kW_0_-300_0kVar	Pref = 0 kW. Set Qref from 0 kVar to 300 kVar, hold for a few seconds, then -300 kVar to 0 kVar	10	105

VAR_0kW_0_-600_0kVar	Pref = 0 kW. Set Qref from 0 kVar to 600 kVar, hold for a few seconds, then -600 kVar to 0 kVar	10	105
VAR_0kW_0_-800_0kVar	Pref = 0 kW. Set Qref from 0 kVar to 800 kVar, hold for a few seconds, then -800 kVar to 0 kVar	10	105
VAR_0kW_0_-1200_0kVar	Pref = 0 kW. Set Qref from 0 kVar to 1200 kVar, hold for a few seconds, then -1200 kVar to 0 kVar	10	105
VAR_0kW_0_+900_0kVar	Pref = 0 kW. Set Qref from 0 kVar to +900 kVar, hold for a few seconds, then +900 kVar to 0 kVar	10	66
VAR_400kW_0_-800_0kVar	Pref = 400 kW. Set Qref from 0 kVar to -800 kVar, hold for a few seconds, then -800 kVar to 0 kVar	10	110
VAR_400kW_0_-1200_0kVar	Pref = 400 kW. Set Qref from 0 kVar to -1200 kVar, hold for a few seconds, then -1200 kVar to 0 kVar	10	110
VAR_400kW_0_+600_0kVar	Pref = 400 kW. Set Qref from 0 kVar to +600 kVar, hold for a few seconds, then +600 kVar to 0 kVar	10	110
VAR_400kW_0_+800_0kVar	Pref = 400 kW. Set Qref from 0 kVar to +800 kVar, hold for a few seconds, then +800 kVar to 0 kVar	10	90
VAR_400kW_0_+1200_0kVar	Pref = 400 kW. Set Qref from 0 kVar to +1200 kVar, hold for a few seconds, then +1200 kVar to 0 kVar	10	100

Table 10. BESS Dataset List - Volts Test

Test name	Scenarios	Sampling Time (ms)	Time Duration (s)
Volts_0kW_Vgrid_100_95_100	Pref = 0 kW. Set the grid voltage from 100% to 95%, hold for a few seconds, then from 95% to 100%	10	105
Volts_0kW_Vgrid_100_90_100	Pref = 0 kW. Set the grid voltage from 100% to 90%, hold for a few seconds, then from 90% to 100%	10	105
Volts_400kW_Vgrid_100_90_100	Pref = 400 kW. Set the grid voltage from 100% to 90%, hold for a few seconds, then from 90% to 100%	10	105

Test Result: Battery Charging

One battery charging sample from the scenarios listed in Table 8 as test P_-426_-210kW is shown in Figure 9. This sample datasets start with Pref from -426 kW to -210 kW, with Qref set to be 0 kVar. The transition of events was observed around 17 seconds.

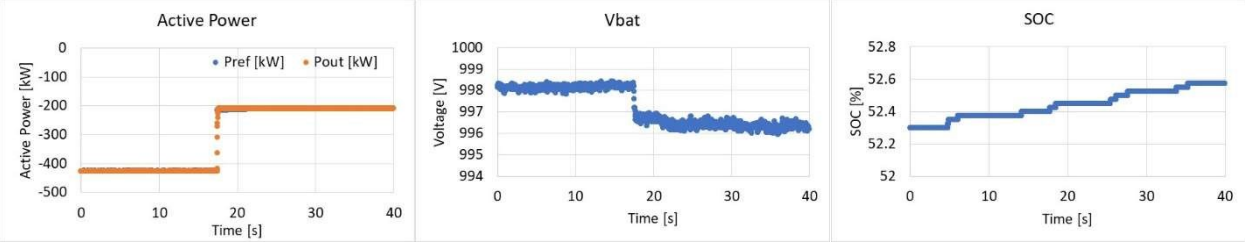


Figure 9. Battery Power (Ref vs. Output), Voltage, and State of Charge (SOC) in Charging Test

Test Result: Battery Discharging

One battery discharging sample from the scenarios listed in Table 8 as test P_+600_+800kW is shown in Figure 10. This sample datasets start with Pref from 600 kW to 800 kW, with Qref set to be 0 kVar. The transition of events was observed around 22 seconds.

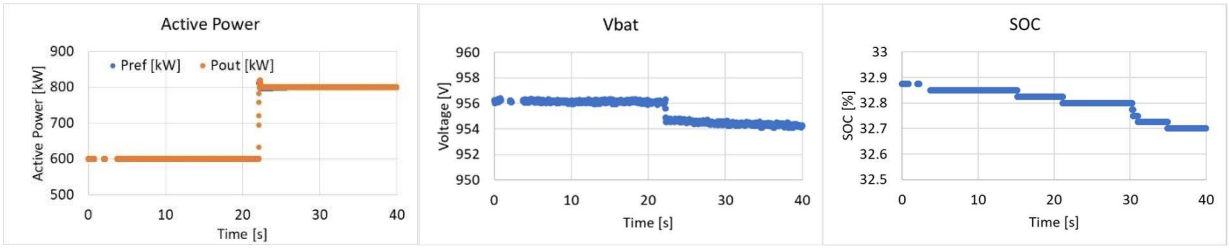


Figure 10. Battery Power (Ref vs. Output), Voltage, and State of Charge (SOC) in Discharging Test

Test Result: Reactive Power Step Test

One reactive power step test sample from the scenarios listed in Table 9. BESS Dataset List - VAR Step Test as test VAR_0kW_0_-300_0kVar is shown in Figure 11. This sample datasets start with Qref from 0 kVar to -300 kVar, hold for a few seconds, then -300 kVar to 0 kVar, with Qref set to be 0 kVar. The transitions of events were observed around 30 and 60 seconds respectively.

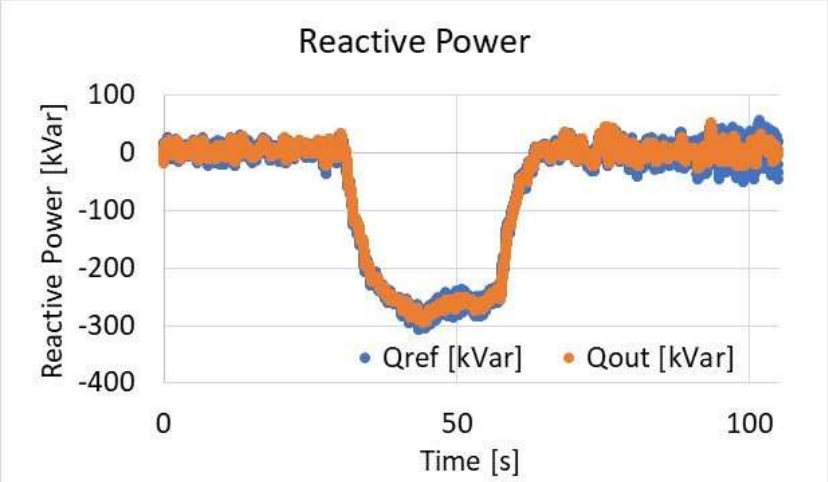


Figure 11. Reactive Power (Ref vs. Output) in Reactive Power Step Test

4.1.3 Milestones Completed

- PV Inverter historical datasets: >20 historical field or laboratory datasets suitable for training and validation obtained.
- PV Inverter Digital Twin model accuracy: ≥90% prediction accuracy during steady-state and transient operation. Residuals have approximate mean = 0 with white noise distribution.
- BESS historical datasets: >20 historical field or laboratory datasets suitable for training and validation obtained.
- BESS Digital Twin model accuracy: ≥90% prediction accuracy during steady-state and transient operation. Residuals have approximate mean = 0 with white noise distribution.

4.2 Learning Framework for Sensor Attack Detection (Task 2)

Model & Simulator Selection (Subtask 2.1**)**
Outcome: Models of existing PV inverter. Reduced order models and higher fidelity models were provided. A 1 kVA, grid connected three-phase inverter with a phase-lock-loop and a closed-loop controller served as the model target. The inverter was grid connected and provided voltage and frequency regulation; it also supports active and reactive power feeding to the grid.

For the complete inverter model, each component in the 1 kVA inverter was modeled as an averaged state-space model:

- grid-connected three-phase inverter operated under sine-triangle pulse-width-modulation (PWM) and both ABC and d-q reference frames
- grid-connected three-phase inverter operated under d-q reference frame
- phase-locked loop (PLL)
- I_{Ld} and I_{Lq} reference generation for a grid-connected three-phase inverter operated under the d-q reference frame

PLECS simulations were developed to simulate each system. A comparison between the PLECS simulation results and the state-space demonstrated the accuracy of the models

.

4.2.1.1 Complete inverter model

With the models developed for each component in the system, the complete inverter model with closed-loop control was developed.

$$\dot{x} = Ax + R(x, u). \tag{15}$$

In (15), x is the inverter state-space variable matrix given by

$$x = \begin{bmatrix} \theta \\ \emptyset_{PLL} \\ i_{Ld}^{*} \\ i_{Lq}^{*} \\ q_{3Ld} \\ q_{3Lq} \\ q_{Ld}^{err} \\ q_{Lq}^{err} \\ i_{Ld} \\ i_{Lq} \\ i_{Lo} \\ v_{Cd} \\ v_{Cq} \\ v_{Co} \\ i_{od} \\ i_{oq} \\ i_{oo} \end{bmatrix}. \tag{16}$$

The state-space system matrix A can be derived as:

```
A = [   0   ki_PLL  0              0              0      0      0   0   0      0      0
        0   0       0              0              0      0      0   0   0      0      0
        0   0       (-sqrt(2)*wc)  0              wc^2   0      0   0   0      0      0
        0   0       0              (-sqrt(2)*wc)  0      wc^2   0   0   0      0      0
        0   0       -1             0              0      0      0   0   1      0      0
        0   0       0              -1             0      0      0   0   0      1      0
        0   0       1              0              0      0      0   0   -1     0      0
        0   0       0              1              0      0      0   0   0      -1     0
        0   0       0              0              0      0      0   0   (-R/L)  0      0
        0   0       0              0              0      0      0   0   0      (-R/L)  0
        0   0       0              0              0      0      0   0   0      0      ((-3*Rg-R)/L)
        0   0       0              0              0      0      0   0   (1/C)  0      0
        0   0       0              0              0      0      0   0   0      (1/C)  0
        0   0       0              0              0      0      0   0   0      0      (1/C)
        0   0       0              0              0      0      0   0   0      0      0
        0   0       0              0              0      0      0   0   0      0      0
        0   0       0              0              0      0      0   0   0      0      0
```

…

```
        0           0           0              0              0              0
        0           0           0              0              0              0
        0           0           0              0              0              0
        0           0           0              0              0              0
        0           0           0              -1             0              0
        0           0           0              0              -1             0
        0           0           0              0              0              0
        0           0           0              0              0              0
        (-1/L)      0           0              0              0              0
        0           (-1/L)      0              0              0              0
        0           0           (-1/L)         0              0              (3*Rg/L)
        0           0           0              (-1/C)         0              0
        0           0           0              0              (-1/C)         0
        0           0           0              0              0              (-1/C)
        (1/Lcoup)   0           0              (-Rcoup/Lcoup) 0              0
        0           (1/Lcoup)   0              0              (-Rcoup/Lcoup) 0
        0           0           (-3*Rg/Lcoup)  0              0              ((-3*Rg-Rcoup)/Lcoup)   ];
```

The R(x, u) term captures nonlinearities and time-varying components in the inverter system that cannot be described by the linear system matrix A.

$$R(x,u)=\begin{bmatrix} K_p^{PLL}v_{oq} \\ v_{oq} \\ 0 \\ 0 \\ \frac{v_{od}P^*-v_{oq}Q^*}{v_{od}^2+v_{oq}^2} \\ \frac{-v_{od}P^*-v_{oq}Q^*}{v_{od}^2+v_{oq}^2} \\ 0 \\ 0 \\ \frac{1}{L}v_{id}+\omega i_{Lq} \\ \frac{1}{L}v_{iq}-\omega i_{Ld} \\ 0 \\ \omega v_{Cq} \\ -\omega v_{Cd} \\ 0 \\ -\frac{1}{L_{coup}}v_{od}+\square\square i_{oq} \\ -\frac{1}{L_{coup}}v_{oq}-\omega i_{od} \\ 0 \end{bmatrix}. \qquad (17)$$

To validate the derived inverter model, a simulation of the state-space model for an inverter with 10 kW real power and 2 kVar reactive power was conducted, (Figure 12, output grid current waveforms). Deviation between model and PLECS simulation (experimentally validated) were negligible.

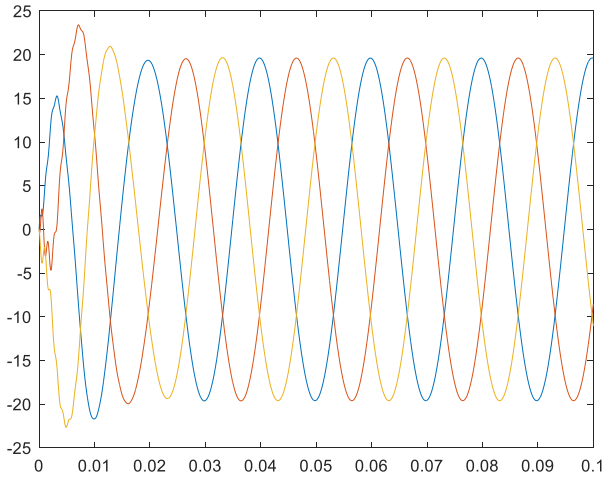


Figure 12. The three-phase grid current for an inverter with 240 V 3-phase grid, 1000 V dc input voltage, 10 kW real power, and 2 kVar reactive power operation using the derived complete inverter state-space model. Each color represents a different phase; model output plotted over PLECS output (no observable difference).

Further it can be seen from Figure 13 (output grid current waveforms) that the derived model represents the step response of the inverter system with closed-loop control accurately as the deviation between the model and PLECS simulation (experimentally validated) were negligible.

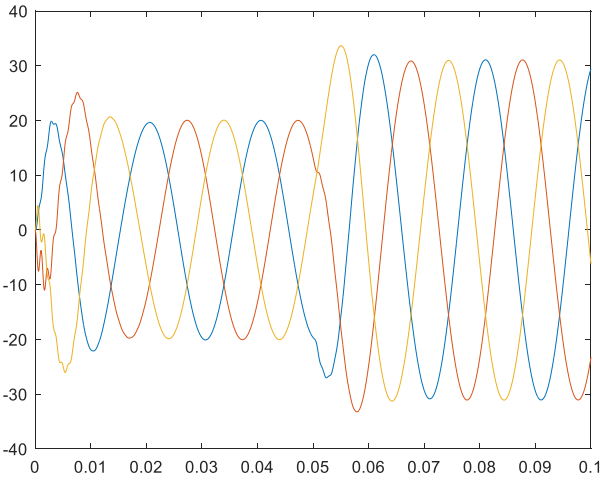


Figure 13. The three-phase grid current for an inverter with 240 V 3-phase grid and 1000 V dc input voltage to have a step change in real power from 10 kW to 15 kW and reactive power from 2 kVar to 5 kVar using the derived complete inverter state-space model. Each color represents a different phase; model output plotted over PLECS output (no observable difference).

4.2.1.2 Summary

The team developed the averaged state-space-based models for PV and BESS inverters with closed-loop control in the d-q frame. A comparison between model-

based simulation results and actual PLECS switching circuit simulation results was done to validate the derived models.

4.2.2 Learning in Simulation (Subtask 2.2)

Outcome: A framework for a Model-assisted Deep Reinforcement Learning agent validated using the simulation setup.

4.2.2.1 Evaluation of State-of-the-Art ML Architectures

Modern ML architectures were tuned to establish baseline performance; specifically, we:

- Identified the reinforcement learning algorithms that are applicable to be used for continuous action space.
- Compared the performance of the identified methods with the Twin-Delayed DDPG implementation.
- When state space cannot be controlled, improved the performance of the machine learning (ML) agent that is used to predict the next state of the plant with higher accuracy compared to the digital twin.
    - Optimized the current FNN architecture parameters
    - Optimized the other hyperparameters
- Implemented a recurrent neural network (RNN) architecture and compared the performance to the FNN.

Deep Deterministic Policy Gradient

The system model for Model-Assisted Machine Learning (MAML) is shown in Figure 14. The digital twin replicates the functionality of the real plant with a linear model. We started with the case that we can control the state space. Later, we switched to the case that we cannot control the state space and we perform the prediction only. We used the DDPG (Deep Deterministic Policy Gradient) algorithm; a model-free off-policy algorithm that can learn continuous actions by integrating DPG (Deterministic Policy Gradient) and DQN (Deep Q-Network) policies.

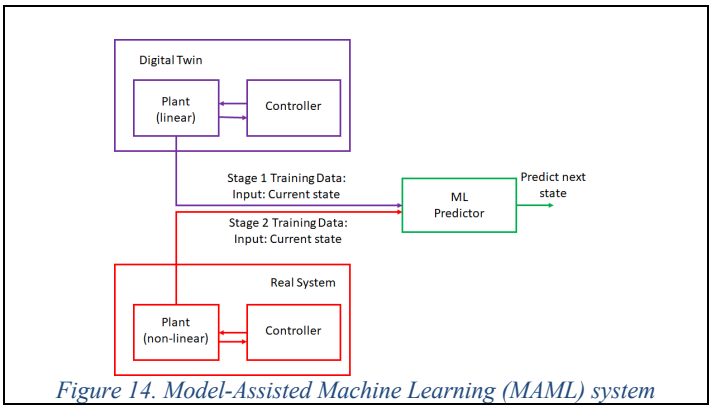


Figure 14. Model-Assisted Machine Learning (MAML) system

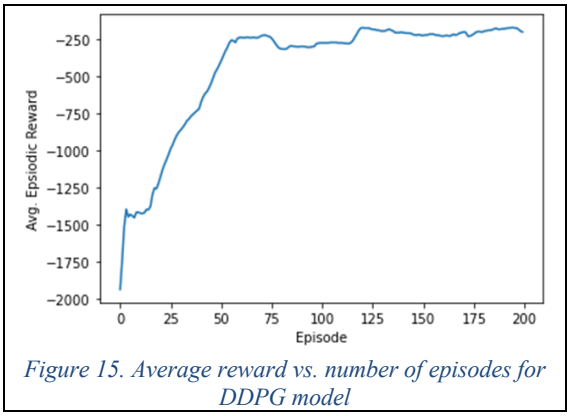


Figure 15. Average reward vs. number of episodes for DDPG model

Although we observed that DDPG achieved good learning performance, it is known to be brittle with respect to the choices of hyperparameters. To address this issue, Twin Delayed DDPG (TD3) was utilized. TD3 was found to increase the reward beyond the performance of DDPG; however, it was found to converge more slowly.

Soft Actor Critic

For the case when we could not control the state space, we switched to the prediction problem. A predictor network with a feedforward neural network (FNN) architecture that takes the current state information from linear and non-linear models and try to predict the next state of the Real System.

Below are the performance results with different number of layers and neurons in each layer, while the loss between the digital twin (black box model) and the true model is 0.154. Two layers with 512 neurons at each layer provides the minimum loss between the predictor and the true model.

# layers & neurons	Loss between the predictor and the true model
512,512	0.015
128,128	0.018
32,32	0.023
8,8	0.027
4,4	0.035
512	0.023
128	0.024
32	0.026
8	0.036
4	0.042

Next, we measured the run time for various number of layers and neurons. The change in the inference time is negligible with increasing number of layers and

neurons. On the other hand, Stage 1 training takes significantly longer time with increasing number of layers and neurons.

	Run time (sec)		
# layers & neurons	Stage 1 training	Stage 2 training	Inference
4	6.59	1.07	0.19
512,512	25.93	2.42	0.20

4.2.2.2 Novel MAML Architectures

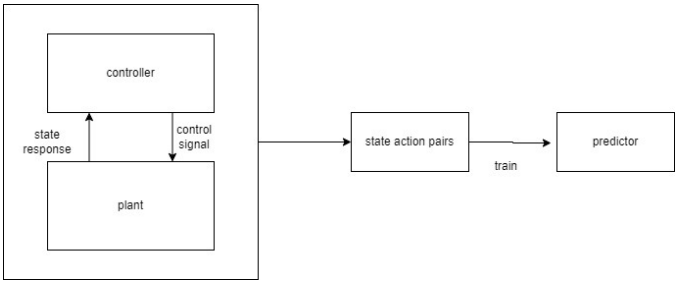


Figure 16. Structure of traditional ML approaches

We modified the structure of the traditional predictor approach (Figure 16) to follow (Figure 17). Specifically, we slice the original state space of the prediction window into multiple intervals to lower the complexity of learning; each agent being responsible for only predicting the behavior at its interval.

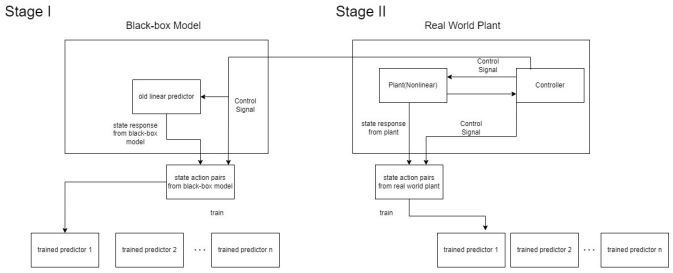


Figure 17. Structure of model-assisted, machine learning (MAML) approach

For the simplicity of evaluation, we used the OpenAI gym simulator for the physical plant to be predicted. OpenAI gym is a simulation environment that is widely used among researchers. Three predictor agents were used for each plant, each being assigned to different intervals of the plant's relevant state variable(s).

Plant	Prediction Intervals	Observation Space (dimensions)	Action Space	Goal	Failure Condition(s)
Cart Pole	Horizontal position [-5, -2), [-2,2), [2,5]	4	Push cart left/right	Minimize deviation (cart at position zero; pole orthogonal to x-axis)	Pole angle greater than +/- 12°; cart outside of viewport
Lunar Lander	Vertical position [10, 6), [6,3), [3,0]	8	Nothing; fire left/right orientation engine; fire main engine	Minimize deviation (lander at position 0,0)	Crash, outside of viewport, no movement
Rotating triple-mass-spring	Motor one angles [−∞,0.5), [0.5,3), [3,∞]	6	Stepper motor positions (2)	Masses at rest (no rotation and constant separation)	Collision and/or rotation of masses

Of the multiple plants we evaluated (Figure 18), the results from the rotating triple-mass-spring were representative. The control task to be solved involves stabilizing freely moving connected masses using a stepper motor.

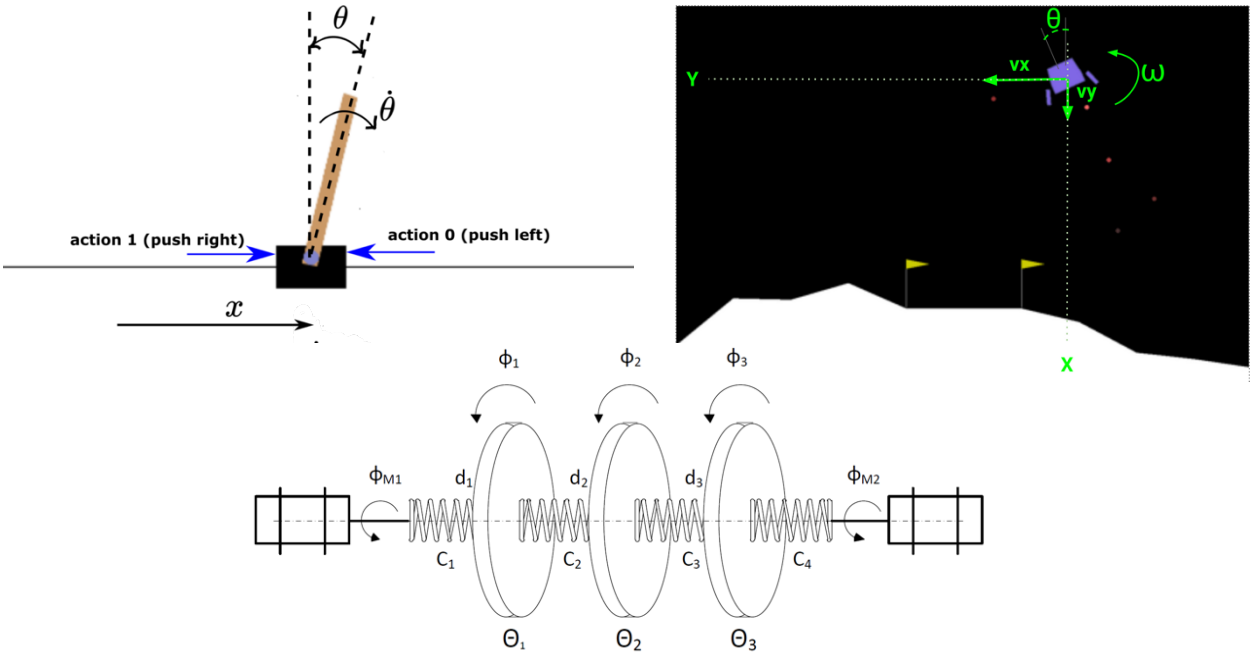


Figure 18. Plants their state variables, and control inputs evaluated using MAML framework. (top left) cart pole, (top right) lunar lander, and (bottom) rotating triple-mass-spring. Images courtesy of Aleksandar Haber, Pau Labarta Bajo, and the Model predictive control python toolbox.

In simulation, the average accuracy of a traditional ML predictor was above 95% (average absolute accuracy for each state), proving that the ML predictor

outperforms the black box model (digital twin). However, even when increasing the number of samples, the performance of MAML did not surpass that of the traditional ML algorithm. We speculate that the reason the MAML approach did not outperform the baseline was due to lack of complexity in the simulated plant behavior.

4.2.3 Milestones Completed

- Models of existing PV inverter completed.
- Models of existing BESS completed.
- PV Inverter MAML accuracy: ≥95% prediction accuracy during steady-state and transient operation. Residuals had approximate mean = 0 with white noise distribution.
- BESS MAML accuracy: ≥95% prediction accuracy during steady-state and transient operation. Residuals had approximate mean = 0 with white noise distribution.

4.3 Grid Modeling (Task 3)

Task interrupted.

4.4 BC Framework (BCF) Development (Task 4)

4.4.1 Security architecture (Subtask 4.1)

Outcome: A BCF that incorporates authentication, encryption, identity management, access control, and message confidentiality and integrity enforcement mechanisms.

4.4.1.1 Overview

We investigated and selected a solution for a private permissioned peer-to-peer ledger and smart contract framework ('BCF'). We looked for a distributed ledger technology that could provide secure communications, identity management, secure updates, network segmentation, and enable automated detection and response using a smart contract framework.

Being a private network, the BCF should offer access control mechanisms and segmented communications channels that allow for communication within a subset of nodes. As a result of a common framework for communication and control, the SBON also serves as a platform for integration and interoperable control of heterogeneous DERs and power systems. The proposed architecture encompasses:

1. Security: authentication, encryption, identity management, access control, and message confidentiality and integrity enforcement mechanisms
2. Communications: secure and resilient network orchestration in a peer-to-peer ledger architecture with transactions regulated by smart contracts.
3. Privacy-preserving data sharing: protected transactions, with validation and verification performed by BC peers, and possibility to collaborate without sharing all raw data.

The implementation of ML-based attack detection at both local and aggregator levels raises privacy concerns. We enabled private channels among trusted partners and provided an environment for collaborative and distributed learning without the need to share the raw data among peers.

4.4.1.2 Approach

In a heterogeneous environment, where peers need to collaborate to achieve a common objective of increasing resilience to malicious attacks, while also protecting their own data and interests, it is natural to seek some structure to guide the interactions among peers. The choice of a ledger to keep track of interactions and contracts to regulate such interactions seems natural in this case. The implementation of such ledger in a distributed manner, using blockchain to validate transactions across distributed nodes, provides inherent data protection resulting from a combination of immutability and transparency. Nodes need not share all their raw data in order to collaborate, satisfying the requirement for data privacy protection.

We evaluated[1] the use of a distributed ledger technology to increase security and resiliency of power systems (outlined below). We created an architecture that accommodates heterogeneous entities, including distributed energy resources (DERs), aggregators, and utility and operator distribution systems that may belong to different organizations (Figure 19). We implemented a private permissioned distributed ledger solution based on Hyperledger Fabric, which provides a flexible framework to manage the interactions of such entities.

[1] Validation was performed through comparison of logged data, program output, and expected results (including hashes of messages, consensus outcomes, etc.). Due to space limitations such data are excluded from the present report though the evaluation procedure itself is described in detail, below.

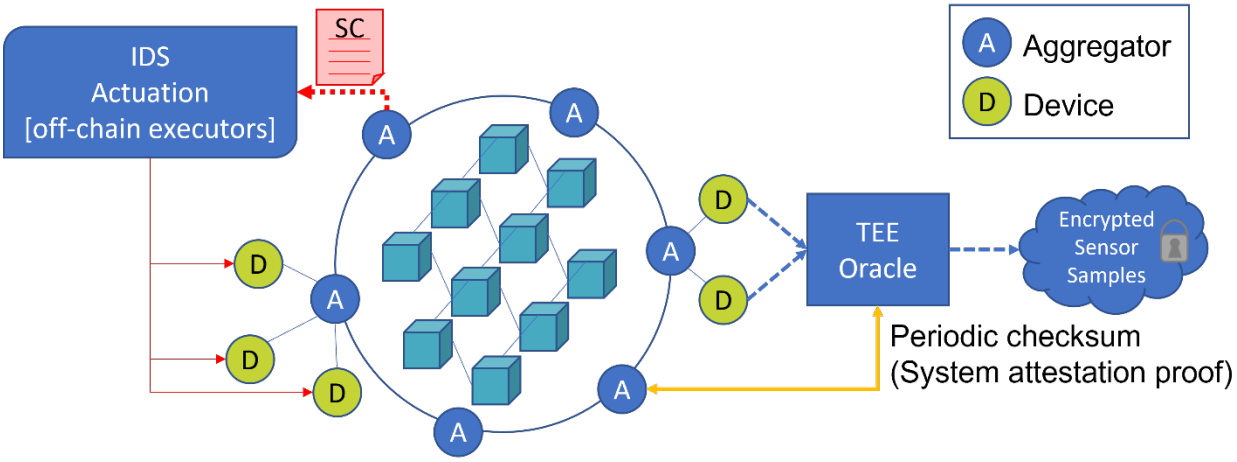


Figure 19. Proposed architecture: aggregators are connected to the BC and guide actuation through smart contracts. DERs are connected to aggregators and also submit sensing measurements to the Trusted Execution Environment. Sensor data is stored in the cloud.

We decided to focus our efforts on Hyperledger Fabric, the most popular DLT framework project in the Hyperledger Foundation. Fabric provides a modular architecture for the development of applications, suitable for application in large scale. The ledger comprises two parts, the world state database, where the current state of the ledger is stored, and the blockchain part, where the records of all transactions are kept.

The ledger state is (by default) stored as key-value pairs (e.g., {DER1: value}) and the blockchain keeps track of all the transactions that caused changes to the state. A transaction can be initiated by any peer in the network, but only fully endorsed transactions (approved by sufficient number of endorser nodes) will actually change the world state. The world state can be re-generated from the blockchain at any time, but it provides a faster and more convenient way to store and query the information in the ledger. Fabric offers some flexibility with the choice of database to store the state, allowing for more complex queries by departing from the default key-value storage to a document object storage allowing data to be stored in JSON format. As a result, queries can be formulated using values of the data, instead of simple queries based on the keys. We illustrate the ledger components in Figure 20.

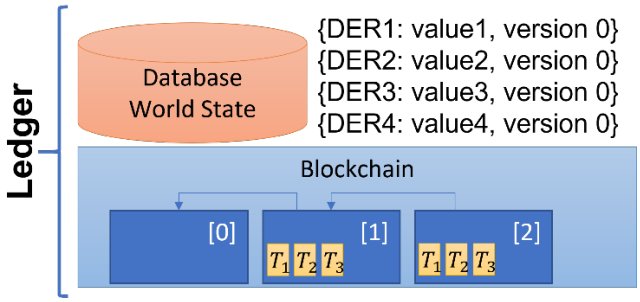


Figure 20. Ledger comprises blockchain and state database.

In Hyperledger Fabric, the recommended consensus protocol is Raft (as of version 1.4.1), a permissioned voting-based mechanism where a leader is elected among ordering nodes to perform ordering [24]. The algorithm is fast, and finality is achieved in a matter of seconds. Raft provides crash fault tolerance, operating under node loss as long as there is majority quorum. In a production network, it is recommended that nodes are spread across different locations to increase resiliency. Fabric offers Crash Fault Tolerance (CFT), that is the ability to continue to operate and reach consensus even if a node is compromised or disconnected. Fabric does not offer native Byzantine Fault Tolerance (BFT), which is the ability to reach consensus when a faulty node is still connected and participating in transactions (a Byzantine node may be a malfunctioning node or a malicious node). The lack of BFT support is a limitation to be mitigated with the use of external libraries. Raft is not BFT, but its design was a step towards implementing BFT for Fabric.

4.4.1.3 Evaluation

Our initial implementation and testing focused on Fabric, setting up and deploying the test network as described in the Fabric Tutorials [25] using the Fabric Gateway to coordinate the actions by clients to submit transactions and query the ledger state. The default configuration of the test network includes two peer organizations, labeled as Org1 and Org2, with one peer per organization to validate transactions, and an additional organization to maintain the Raft ordering service, which decides on the order of the transactions based on a deterministic consensus algorithm and submits them to the blocks. The blocks are distributed to the “peer” nodes, representing DERs and aggregators, and then added to the ledger.

We used a script to generate the test network as described in Fabric Tutorials [25]. The test network had two organizations, Org1 and Org2, and an additional organization to maintain the ordering of service. We added peers and ordering nodes to the network as needed to represent the entities in the scenarios of interest. Peers in the network were able to validate transactions, and ordering nodes reached consensus on the order of transactions and submitted them to the blocks. The blocks were distributed to peers and then added to the ledger.

One important and attractive feature of Fabric is the flexibility to develop a network of networks with a certain degree of trust, in which organizations may be supported by a private channel to communicate and perform transactions, where such transactions are “invisible” to the rest of the network. Channel layers of communication are exclusive to invited members who are authenticated and authorized to transact on the channel and are defined by member organizations, anchor peers per member that serve as endpoints for discovery and communication, a shared ledger distinct to each channel, chaincode application(s), and the ordering service node(s). Orderers play an important role as they enforce basic access control for the channel.

The first block on the chain, called the genesis block, stores configuration information about the channel policies, members, and anchor peers. In our test network, we created a new channel called 'mychannel' and joined each peer in Org1 and Org2.

Smart contracts that contain the business logic to interact with the channel ledger are deployed in packages referred to as chaincode. We installed the chaincode on the peers of each organization and then deployed it to the channel so that it could endorse transactions and interact with the ledger. We experimented with asset-transfer-basic chaincode example already provided with the distribution, which was successful based on logs and program output.

After the network was setup as indicated, we then created a test harness to assess the various transactional operations for placement, observation, and confirmation on the blockchain. In particular, we created an initial data schema to include a unique asset ID, owner information, and integral data to simulate generic sensor measurement data. We then populated the ledger and subsequently modified the

data and observed the transactions through various types of successful queries on the submitting node. We confirmed these transactions through more queries, this time on a different node.

We then successfully verified the adaptability of our test network by adding a new node, joining that peer to the existing channel that we created earlier, and then installing the chaincode. We validated its association with the channel using an appropriate query from the node itself, where, again log files and output were used for confirmation.

Finally, we added functionality to the data schema itself by including a verifiable hash value of the randomly-generated measurement sensor data for a randomly-generated DER owner. These values were verified using the `md5sum` tool.

4.4.2 Communications architecture (Subtask 4.2)

Outcome: Developed methods for secure and resilient network orchestration, identity management, and security automation using a peer-to-peer ledger architecture and smart contract framework.

The architecture illustrated in Figure 19 achieves the objectives of both Subtasks 4.1 and 4.2. The solution tools are open-source, and support the necessary customization for data management, actuation control, and interactions among the enterprises in a regulated manner that keeps records of every transaction. The initial implementation has tremendous potential to attend the needs of power grid application, as well as many other applications that include distributed data collection and even distributed learning.

4.4.3 Privacy-Preserving attack detection R&D (Subtask 4.3)

Outcome: Developed algorithms using homomorphic encryption to enable privacy-preserving collaborative attack detection for local and global levels.

We assumed some producers (a minority) were controlled by an attacker; thus, these producers are malicious and inject power which is not consistent with the promised one. We illustrate the system model with attack in Figure 21. Furthermore, if producers are required to report some measurements/data to the aggregator (to

monitor producers' measurements), the malicious producers may intelligently report bad data to avoid being detected.

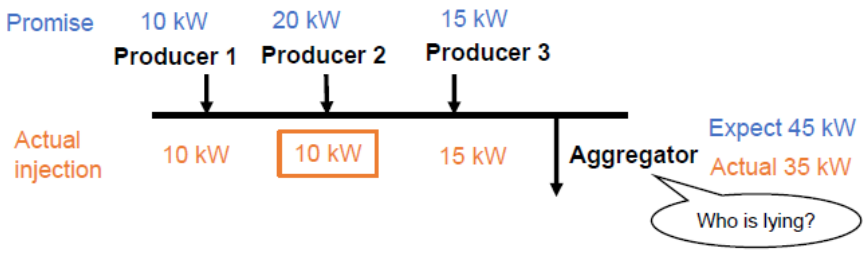


Figure 21. System Model

4.4.3.1 In the non-private attack detection algorithm [26], the aggregator needs the information of estimated and sensed states for t = 0, 1, 2, … to iteratively compute *w(t)* and *r(t), respectively*. If r(t) = 0 for all t, then there is no attack; otherwise, an exists attack. For the privacy-preserving attack detection and identification we protected the privacy of producers' state and measurement (*w(t)* and *y(t)*), but still enabled the aggregator to correctly compute detection results.

Security and Privacy Analysis:
Integrity. The linearity of the (non-private) detection and identification algorithms guarantees the computation of the private algorithms can be done locally. Thus, malicious producers cannot modify honest producers' shares. Also, robust reconstruction guarantees the malicious minority cannot change the final detection and identification results by submitting malicious/ wrong shares.

Privacy. Following the perfect secrecy of Shamir's secret sharing, the private algorithm leaks nothing except what can be learned from the final results. Then, we analyzed what can be learned from the final results. We considered the following cases.

Case 1. No measurement noise. $r(t) \rightarrow 0$ for $t \rightarrow \infty$ If and only if there is no attack, which indicates that an honest producer's measurement is independent of the released result I($r(t) < \delta$) for a large enough t. Thus, there is no privacy leakage for honest producers.

Case 2. The measurement noise is independent of measurement y(t) for t = 0, 1,... Then, due to the linearity of r(t), we can represent the residual in this case by $\tilde{r}(t) = r(t) + N(0, \sigma^2)$, where an honest producer's measurement is in dependent of both

I(|r(t)| < δ) (for a large enough t) and N (0, σ^2), and thus in dependent of the released result I(|r˜(t)| < δ). Therefore, there is no privacy leakage for honest producers.

An IEEE 14 bus system was used for evaluation with the following values.

$$A = \begin{bmatrix} -1.52 & -327.93 & 0 & 256.18 & \cdots \\ 1 & 0 & 0 & 0 & \cdots \\ 0 & 256.18 & -1.52 & -509.68 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}_{19\times 19}, \quad C = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}_{19\times 19}$$

Gaussian noise was added in x(t) and y(t) as state and measurement noise. Figure 22 and Figure 23 show the results of state x(t), estimated state w(t), measurement y(t), and detection residual r(t). When there is no attack, the detection residual r(t) = 0 (ignoring the random perturbation of state/measurement). When there exist attacks (either to x or y), the detection residual r(t) becomes non-zero after the attack start.

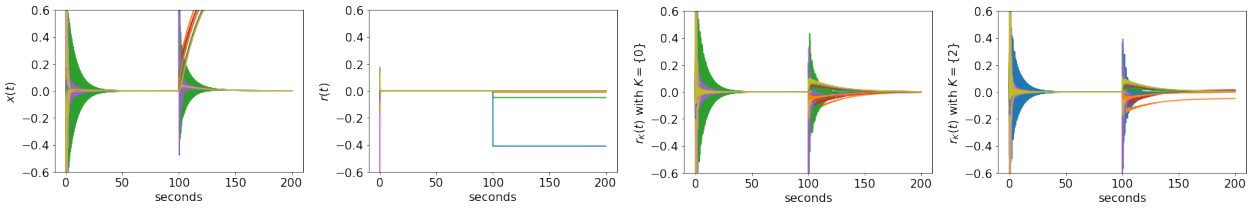


Figure 22. Attack Detection/Identification in IEEE 14 Bus (when true attack set $K_* = \{0\}$)

Figure 22 and Figure 23 show the state change (i.e., $x(t)$), detection result (i.e., $r(t)$), and identification results (i.e., $r_K(t)$ for K = {0} or {2}) under two different attacks (K^* = {0} or {2}). The results are consistent with the theorem. Recall that the identification residual $r_K(t) = 0$ if and only if the true attack set is covered by the assumed attack set, i.e., $K^* \subseteq K$. Thus, our experimental results show that the privacy-preserving version can obtain the same result as the non-private one (ignoring negligible errors due to field conversion).

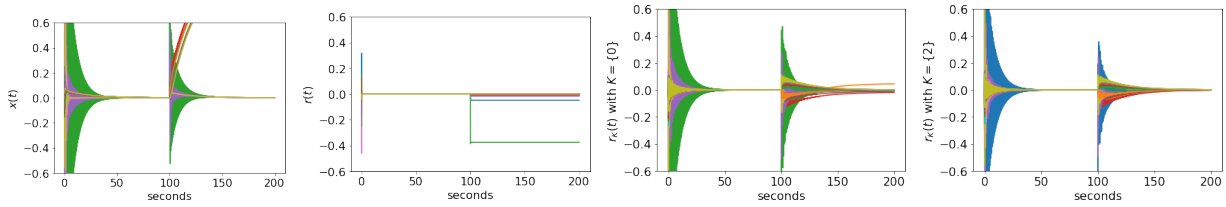


Figure 23. Attack Detection/Identification in IEEE 14 Bus.

4.4.4 Milestones Completed

- Functional security and communications architecture specified.
- Communications restrictions/availability for trusted and untrusted nodes.
- For local (DER) level false-data injection attack detection: detection accuracy, privacy (confidentiality), and computation efficiency (run time).
- For global (aggregator or utility) level attack detection: detection accuracy, privacy level (privacy budget), and computation efficiency in terms of run time.

4.5 Enabling Physical World Access from Blockchain (Task 5)

4.5.1 Theoretical Foundations (Subtask 5.1)

Outcome: Developed the theoretical foundation for off-chain execution.

4.5.1.1 Motivation

The security of Blockchain relies on the honest majority, yet, for public blockchain, this makes the assumption that all the participants are rational. More efficient consensus mechanisms to achieve consensus in private/semi-public setting are available. However, regardless of the consensus mechanisms the challenge of limited on-chain computational capability remains a challenge.

4.5.1.2 Approach

We address this problem using off-chain execution. Based on our preliminary survey, there are three types of off-chain execution mechanisms. The first type of off-chain execution directly leverages the trusted execution environment (TEE) at a single node in the network. For example, in towncrier [15], Zhang et al. proposed to use Intel SGX enclave as a trusted entity for external data feed, it essentially executes the data retrieval process via a single trusted node. One limitation of this

approach is that centralization is re-introduced into a supposedly decentralized system. The second approach, on the other hand, leverages a network of computing nodes equipped with Intel SGX trusted execution environment. In Ekiden [16], similar to one on-chain computing, a computation is replicated among many nodes in the blockchain network, and the compute node will reach consensus among all the computation results on which one to commit to as a network. However, due to the replication of the computation, there is still non-trivial overhead for decentralized security.

Furthermore, in order to obtain a reasonable degree of decentralization, it is important that the quorum has a non-trivial number of participating parties. From the execution time perspective, if all the nodes are well connected, then latency may still be manageable, and this response time is important for the context we are studying. The last approach proposed in PrivacyGuard [17] aims to combine remote attestation and smart contracts to achieve off-chain execution.

When the computation takes place off-chain, several challenges occur. First, the correctness of the contract execution can no longer be guaranteed by the blockchain consensus. To this end, they propose ``local consensus'' for the contracting parties to establish trust on the off-chain computation via remote attestations. The main intuition is that remote attestation allows a prover to attest its system state to a remote verifier. If all of the parties involved in a smart contract perform remote attestation on the secure enclave and can verify the system states of the execution environment, then it is sufficient to trust the outcome from that enclave. This approach makes the assumption that the trusted execution environment is free of vulnerability, this may not be correct all the time. Furthermore, the latency from the remote attestation may be non-trivial. It is important to take these factors into consideration in the design of off-chain execution.

Build on top of this systemization effort, we have further developed the universal compossibility for TEE-based off-chain execution to set the foundation for the off-chain execution in the proposed power-grid blockchain. In an off-chain execution, a smart contract’s execution is split into control and computation, where the computation actually takes place off-chain, several challenges occur.

First, the correctness of the contract execution could no longer be guaranteed by blockchain consensus. To this end, we propose “local consensus” for the contracting

parties to establish trust on the off-chain computation via remote attestations. Second, the execution of contract is no longer atomic when the computation part is executed off-chain. We design a multi-step commitment protocol to ensure that result release and data transaction remain an atomic operation, where if the computation results were tampered with, the data transaction would abort gracefully.

We implemented a prototype of off-chain execution using Intel SGX as the TEE technology and Ethereum as the smart contract platform. We chose these two technologies for implementation due to their wide adoption. Our design generally applies to other types of trusted execution environments and blockchain smart contract platforms. The platform fulfills the goal of user-defined data usage control at reasonable costs and we show that it is feasible to perform complex data operations with security and privacy protection as specified by the data contract.

4.5.1.3 Summary

The BCF detailed above necessitates a Trusted execution environment (TEE) executing on a limited resource embedded system (e.g., a microcontroller) to provide cost-effective, local protection for DERs. We systemized existing methods for off-chain execution understanding the limitations and trade-offs of such systems. Research then focused on generating the proof for TEE-based off-chain execution under the universal compossibility framework. The main accomplishment lies in a non-trivial proof, which provided a sound theoretical basis for the utilization of TEE-secured DER as the foundation of a resilient grid.

4.5.2 Trusted Modules for Embedded Processors (Subtask 5.2)

Outcome: We leveraged Root-of-trust for low-cost microcontrollers (MCU) to provide protection for manufacturers seeking to safeguard their valuable machine learning models against intellectual property (IP) theft.

4.5.2.1 Motivation

With recent advances in deep learning (DL) [18], there is a growing need to deploy the machine learning (ML) models on smart microcontrollers (MCUs) at the Edge for communication efficiency and privacy protection. This deployment paradigm on MCUs is often referred to as tiny machine learning (TinyML) [19].

4.5.2.2 Approach

We proposed Secure TinyML (STML) to protect model IP on MCUs under an untrusted software stack based on commercial off-the-shelf hardware. There were two main challenges:

- Constrained Memory. TEE utilizes isolation to safeguard memory contents, but it can lead to memory scarcity for DL inference execution in the secure world and other tasks in the normal world due to the limited available memory resources.
- Co-Optimization. The memory swapping during world switches of TrustZone and the use of cryptographic operations for swapped data protection significantly increase the runtime latency of DL execution.

System Design

STML protects the IP of TinyML models using a system and algorithm co-design approach. As shown in Figure 24, STML consists of an offline optimization engine and a runtime IP protection mechanism. The offline optimization engine outputs a resource allocation strategy to minimize the TinyML task execution delay.

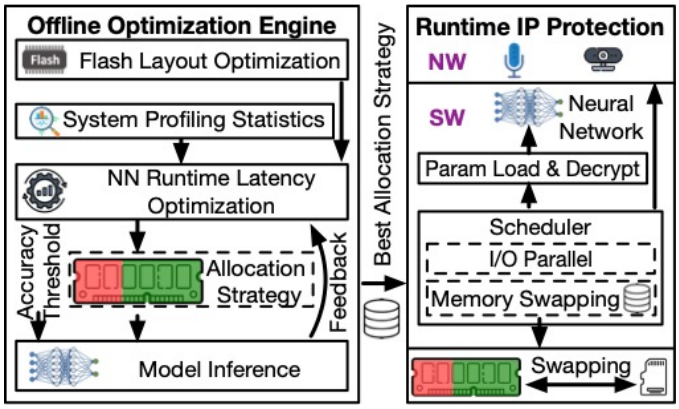


Figure 24. STML System Design

Evaluation

To evaluate the performance of STML on a range of TinyML tasks using various trained models, we evaluated it with models from MLPerf Tiny Benchmark [21] and MicroNets [22]. The benchmark TinyML tasks include keyword spotting (KWS), anomaly detection (AD), visual wake words detection (VWW), and image classification (IC). Table 11 shows measurement data of the used models when all system resources are available on the MCU. We ensured our algorithm-level model optimization adhered to the performance requirements and quality targets specified by MLPerf Tiny Benchmark.

A TinyML task runs in the secure world, while other tasks including LED Toggling, Logging, and AudioSampling, are executed in the normal world. The metadata of these tasks is illustrated in Table 12. Note that the flash size of AudioSample includes both the code size in the internal flash memory the data size in the SD card. Similar to tasks in widely-used cyber-physical systems like ArduPilot [23], these normal world tasks have higher execution priorities, as they are responsible for critical operations.

Table 11. Metadata of TinyML Benchmark Models

Task	Model	Flash (KiB)	RAM (KiB)	Latency (ms)	Metric
Keyword Spotting	DS-CNN	144.80	31.25	81.29	90% (Top-1)
	MicroNet-KWS(S)	192.55	70.14	233.05	
Anomaly Detection	Deep AutoEncoder	328.96	10.57	7.64	0.85 (AUC)
	MicroNet-AD(S)	327.64	120.14	445.66	
Visual Wake Words	MobileNetV1 0.25x	420.07	108.82	256.68	80% (Top-1)
	MicroNet-VWW(S)	363.60	77.71	146.01	
Image Classification	ResNet-8	187.04	62.32	373.33	85% (Top-1)

Table 12. Metadata of Tasks in the Normal World

Task	Flash (KB)	SRAM (KB)	Frequency (Hz)	Priority
LED Toggle	56.90	25.14	50	2
Logging	89.70	41.34	20	3
AudioSample	170.94	114.35	10	1

4.5.2.3 Summary

We introduced STML, a TinyML model IP protection system for MCUs utilizing ARM TrustZone. We proposed a memory swapping scheme to address the limited memory issue and minimize I/O and inference latency through system and algorithm level optimization. Our approach effectively balances memory usage, latency, security, and accuracy, resulting in a 40% reduction in runtime overhead compared to non-optimized solutions. Although initially designed for systems with predictable workloads, STML can be adapted to other systems by adjusting the DL execution latency modeling to accommodate their specific characteristics.

4.5.3 Milestones Completed

- Security modeling was validated based on the percentage of security requirements proved using the security mechanism.

- The performance of the trusted module was found to be more efficient in terms of energy, memory, throughput (e.g., latency) overhead compared to the state of the art.

5. **Significant Accomplishments and Conclusions:** Accurate Digital Twins (DTs) and model-assisted machine learning (MAML) approaches critical for cyber-attack detection for PV Inverter and BESS developed and verified. The PV Inverter and BESS DTs had ≥90% prediction accuracy during steady-state and transient operation and were each based on >20 historical field datasets suitable for training. The MAML approaches achieved ≥95% prediction accuracy during steady-state and transient operation of non-linear physics-based models, including PV Inverter and BESS models. For local, DER-level attack detection, the privacy-preserving attack detector was found to have <5% accuracy degradation compared with the no privacy case; formal privacy guarantees were provided and reasonable run time of <60s for local level detection were achieved.

   Specifications for both functional security and communication architectures for the BCF were completed. The specifications incorporated authentication, encryption, and message confidentiality and integrity enforcement mechanisms in BCF, as well as secure and resilient network orchestration, using a peer-to-peer ledger architecture and smart contract framework. To connect the cyber world that is orchestrated by the blockchain fabric and the physical world in which the power network resides, a trusted machine learning framework was implemented and benchmarked as a trusted module for resource-constrained systems. A performance degradation of <10% was observed compared to the processing without trusted module on average. By implementing the MAML approaches using this framework, individual DER can now securely attest to performing attack detection as an off-chain function, thus proving the use of smart contracts for command and control (C2) of DER .

6. **Path Forward**

   Two avenues of future research and development are suggested based upon project outcomes:

   1. To protect legacy DER the MAML and BC technologies can be integrated into a modular, plug-and-play security module. The module would need to employ a networking interface to receive sensor/status data from, and transmit C2 signals to, DER. Interfacing with the aggregator/utility equipment could be

accomplished using, for example, IEC 61850 to the applicable IEEE 1547-2018 protocol.

2. The SBON can be extended to secure information exchanges (data and C2) between utility distribution controllers, energy aggregators, and DER. Least privilege-based network segmentation for communications between utilities and DER controllers could be achieved using a PKI tailored to provide role-based access control to restricted and securely segmented communications channels within the SBON. Providing a unified and interoperable control interface could be accomplished using a standardized data model based on multiple standardized DER and grid control frameworks to allow for low-cost, low-effort integration and interoperable control of heterogeneous DER and power systems.

7. Products

Jinwen Wang, Yuhao Wu, Han Liu, Bo Yuan, Roger D. Chamberlain, and Ning Zhang, “IP Protection in TinyML,” in Proc. of 60th ACM/IEEE Design Automation Conference (DAC), July 2023.

8. Project Team and Roles

(e.g., DOE personnel, students, collaborating organizations).

Name	Institution	Role	Contribution
Adams, Stephen	VT	Investigator	Blockchain
Chen, Zhe	GER	Investigator	Digital Twins
Erpek, Tugba	VT	Investigator	MAML
Florez, Orlando	VT	Program Manager	Budgets & Schedule
Gerdes, Ryan M.	VT	Principal Investigator	MAML
Giani, Annarita	GER	Investigator	Digital Twins and Blockchain
Gu, Patrick	VT	Student	MAML
Gu, Xiaolan	UA	Student	Privacy-preserving Attack Detection
Heaslip, Kevin	VT	Investigator	Integration & Coordination
Li, Ming	UA	Investigator	Privacy-preserving Attack Detection
Morales-Rodriguez, Marissa	DOE	Technology Manager	Supervisory
Sagduyu, Yalin	VT	Investigator	MAML
Salasoo, Lembit	GER	Program Manager	Digital Twins
Skinner, Tucker	USU	Student	PV Inverter Models
Wang, Chun-Tao	VT	Student	MAML
Wang, Hongjie	USU	Investigator	PV Inverter Models
Wang, Jinwen	WUSTL	Student	Off-chain Execution

Zhang, Ning	WUSTL	Investigator	Blockchain & Off-chain Execution

9. References

[1] C. Kwon, W. Liu, and I. Hwang, “Security analysis for cyber-physical systems against stealthy deception attacks,” in 2013 American control conference, pp. 3344–3349, IEEE, 2013.

[2] N. Zhang, J. Li, W. Lou, and Y. T. Hou, “Privacyguard: Enforcing private data usage with blockchain and attested execution,” in Data Privacy Management, Cryptocurrencies and Blockchain Technology, 2018.

[3] F. Pasqualetti et al., “Attack detection and identification in cyber-physical systems,” Transactions on automatic control, 2013.

[4] S. Sridhar et al., “Model-based attack detection and mitigation for automatic generation control,” Transactions on Smart Grid, 2014.

[5] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, “Optimal and autonomous control using reinforcement learning,” IEEE trans. on neural net. and learning sys., vol. 29, no. 6, pp. 2042–2062, 2018.

[6] K. G. Vamvoudakis, F. R. Pour Safaei, and J. P. Hespanha, “Robust event-triggered output feedback learning algorithm for voltage source inverters with unknown load and parameter variations,” International Journal of Robust and Nonlinear Control.

[7] A. Kanellopoulos and K. G. Vamvoudakis, “Non-equilibrium dynamic games and cyber–physical security: A cognitive hierarchy approach,” Systems & Control Letters, vol. 125, pp. 59–66, 2019.

[8] J. Buckman, D. Hafner, G. Tucker, E. Brevdo, and H. Lee, “Sample-efficient reinforcement learning with stochastic ensemble value expansion,” in Adv. in Neural Information Proc. Sys., pp. 8224–8234, 2018.

[9] S. Gu, T. Lillicrap, I. Sutskever, and S. Levine, “Continuous deep q-learning with model-based acceleration,” in ICML, 2016.

[10] A. Kosba et al., “Hawk: The blockchain model of cryptography and privacy-preserving smart contracts,” in IEEE S&P, 2016.

[11] D. Energy, “Increasing distribution system resiliency using flexible der and microgrid assets enabled by openfmb,” US DOE Grid Modernization Initiative, 2018.

[12] Iuon-Chang Lin, Tzu-Chun Liao, “A survey of blockchain security issues and challenges,” vol. 19, pp. 653–659, September 2017.

[13] B. Herman, “PBFT Consensus,” in Sawtooth RFCs, Hyperledger, June 2018.

[14] E. Mengelkamp, B. Notheisen, C. Beer, D. Dauer, and C. Weinhardt, “A blockchain-based smart grid: towards sustainable local energy markets,” Computer Science-Research and Development, 2018.

[15] F. Zhang, E. Cecchetti, K. Croman, A. Juels, and E. Shi, “Town crier: An authenticated data feed for smart contracts,” in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 270–282, ACM, 2016.
[16] R. Cheng, F. Zhang, J. Kos, W. He, N. Hynes, N. Johnson, A. Juels, A. Miller, and D. Song, “Ekiden: A platform for confidentialitypreserving, trustworthy, and performant smart contracts,” in 2019 IEEE European Symposium on Security and Privacy (EuroS&P), pp. 185–200, IEEE, 2019.
[17] Y. Xiao, N. Zhang, J. Li, W. Lou, and Y. T. Hou, “Privacyguard: Enforcing private data usage control with blockchain and attested off-chain contract execution,” in European Symposium on Research in Computer Security, pp. 610–629, Springer, 2020.
[18] Y. LeCun et al., “Deep learning,” Nature, 2015.
[19] P. Warden et al., TinyML. O’Reilly Media, 2019.
[20] J. Lin et al., “Mcunet: Tiny deep learning on iot devices,” in NeurIPS, PMLR, 2020.
[21] C. Banbury et al., “MLPerf tiny benchmark,” arXiv, 2021.
[22] C. Banbury et al., “Micronets: Neural network architectures for deploying tinyml applications on commodity microcontrollers,” MLSys, 2021.
[23] “Ardupilot.” https://ardupilot.org/.
[24] Hyperledger Foundation, Hyperledger Fabric Documentation. [Online]: https://hyperledger-fabric.readthedocs.io/en/latest/orderer/ordering_service.html#ordering-service-implementations
[25] Hyperledger Foundation, Hyperledger Fabric Tutorials. [Online]: https://hyperledger-fabric.readthedocs.io/en/latest/tutorials.html.
[26] Pasqualetti, Fabio, Florian Dörfler, and Francesco Bullo. "Attack detection and identification in cyber-physical systems." IEEE transactions on automatic control 58.11 (2013): 2715-2729.