This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2023.3244329

ECS-Grid: Data-Oriented Real-Time Simulation Platform for Cyber-Physical Power Systems

Tianshi Cheng, *Graduate Student Member, IEEE,* Tong Duan, *Member, IEEE,* and Venkata Dinavahi, *Fellow, IEEE*

***Abstract*—ECS-Grid is the first data-oriented real-time electromagnetic transient simulation platform for cyber-physical power systems (CPPS). Traditional simulation tools are constrained by object-oriented programming (OOP) architecture, which is now a significant obstruction to creating a comprehensive cyber-physical simulation. Therefore, the proposed ECS-Grid platform follows a new data-oriented paradigm based on an Entity-Component-System (ECS) framework, which delivers higher flexibility, extensibility, scalability, and performance to support cyber-physical system research. ECS-Grid proposes a layer of virtual intelligent electronic devices (vIEDs) to model IEDs in CPPSs. The vIEDs directly talk to physical components and communicate asynchronously with cyber services via the proposed high-performance JSON-like binary protocol. Tests with the islanding and the man-in-the-middle cyber attack scenarios on a 711-node AC-DC microgrid cluster based on a modified CIGRE 15-Bus system are performed and give accurate results. A faster-than-real-time performance is achieved on the 10th Gen Intel® Core™ i7 computer, and real-time performance is achieved on distributed embedded NVIDIA® Jetson platform. The ECS-Grid design and test results demonstrate the potential of the ECS data-oriented paradigm and may inspire the renovation of industrial simulation software.**

***Index Terms*—Cyber-physical power systems, data-oriented programming, digital twin, entity-component-system, electromagnetic transients, faster-than-real-time, field programmable gate arrays, intelligent electronic devices, real-time systems, microgrids, parallel processing**

I. Introduction

The transition to clean and renewable energy in the power industry is playing a significant role in reducing the emission of greenhouse gases and fighting climate change [1]. However, the traditional power systems that rely on centralized control networks and controllable generators become insufficient to meet the challenges of future power systems such as microgrids with the high penetration of uncertain and unstable renewable energy [2]. Therefore, new intelligent decentralized control solutions based on modern information and communication technologies are emerging to face the new challenges [3]–[5]. The new research works heavily involve communication between control centers and intelligent electronic devices (IEDs), which are the foundation of smart grid and power system automation [6]. Thus, detailed and accurate real-time simulation of cyber-physical power systems (CPPS) [7] is necessary for future power system research [8]–[10]. However, the scalability, flexibility, and performance of the traditional power system analysis tools and communication analysis tools are inadequate. For example, the existing software-based simulation approaches such as EPOCHS [11], GECO [12], INSPIRE [13] and the simulators proposed in [14]–[16] aim to create a network interface for existing power system simulation tools and glue the two domains in power grid simulator and communication network simulators (NS-2, NS-3 and OPNET et al.), which cannot reflect the behaviors of IEDs in real-time environments and must handle the complicated synchronization between two different simulation domains. Some works such as [17]–[20] bridged the commercial real-time power system electromagnetic transient (EMT) digital simulators to the communication simulation systems, which can achieve real-time performance and more realistic behaviors. However, the high-cost commercial EMT simulators were designed for industrial verification purposes and still lack the scalability and flexibility for CPPS-related academic research.

The essential problem is that both power grid and communication network simulation tools are initially designed for their single domain. The traditional software mainly based on object-oriented programming (OOP) has become a huge obstruction to building a native and comprehensive cyber-physical simulation platform. The industrial programs are dealing with various forms of data and their combinations, while the OOP paradigm emphasizes predetermined inner structure and relationships of objects. Plain data such as an array of float numbers can represent many things in the computer world. On the opposite, a class and its object can only be used for one purpose predefined by abstract templates, which brings significant difficulties to repurposing existing designs and thus cannot elegantly describe and solve the problems in the complex interdisciplinary CPPS.

Therefore, this paper proposes for the first time, the ECS-Grid: a novel real-time cyber-physical EMT simulation platform with virtual IEDs (vIEDs) based on the cutting-edge entity-component-system (ECS) software framework. The proposed ECS-Grid simulation platform has the following major advantages:

1) High Flexibility: Compared to the traditional dominating object-oriented paradigm which is based on *Polymorphism, Abstraction, Inheritance, and Encapsulation*, the ECS framework is based on a data-oriented paradigm:

This work is supported by the Natural Science and Engineering Research Council of Canada (NSERC).

Tianshi Cheng and Venkata Dinavahi are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta T6G 2V4, Canada. Email: tcheng1@ualberta.ca, dinavahi@ualberta.ca.

Tong Duan is with the National Digital Switching System Engineering and Technological Research Center, Zhengzhou, China. Email: duantong21@outlook.com.

Entity (usually an integer), *Component* (pure data structure), and *System* (plain functions to perform algorithms on components), where entities are defined by the combination of data components, and component functionalities are defined by systems. This data-oriented paradigm avoids dependency complexities caused by OOP inheritance and brings flexible model description ability. Any data component and the system can be replaced not only at the compiled time but also at the run time. Such a feature is highly desired for cyber-physical simulation since the various form of data and data flows are the major concerns. For example, real-world IEDs are composed by multi-functional circuit boards and these replacable boards can be represented by data components on a vIED entity.

2) High Extensibility: with the advanced data-oriented design, components and systems are grouped into plugins in ECS-Grid, and a simulation application is composed of a set of plugins. In contrast to traditional software which often provides a huge library as an undividable whole, ECS-Grid allows users to only pay for what they need. Although it is initially designed for CPPS simulation, it can run pure physical simulation similar to simulators without cyber layers, or run the cyber features for other purposes without the physical EMT simulation. Moreover, the users can create plugins easily even in dynamic libraries with their customized components and systems, and add or override core functionalities such as the matrix solvers or additional communication protocols.
3) High Scalability: With the benefits from the ECS framework, a vIED layer is proposed which mainly utilizes scalability protocols from the message-oriented asynchronous ZeroMQ [21]. A MessagePack-based [22] JSON-like simulator protocol is proposed for the simulator to bridge various industrial protocols. The utilization of the middleware makes it easy to scale ECS-Grid from a single CPU node to multi-thread applications or even distributed networks which resembles real-world automation systems. The performance test on a single-thread ZeroMQ vIED with MessagePack-based protocol shows a minimal latency of $6\mu s$ and an average latency of 20μs with an effective 60Mbit/s bandwidth, which is quite enough for a wide range of application scenarios.

A 711-node AC/DC microgrid cluster based on the modified CIGRE-15 Bus microgrid system with a man-in-the-middle cyber-attack scenario is set up for demonstration and performance evaluation. The simulation results show that the proposed solution can achieve faster-than-real-time (FTRT) performance on 10th Gen Intel® Core™ i7 CPUs and real-time performance on NVIDIA Jetsons with dual-core ARM v8 CPUs.

The paper is organized as the following: Section II introduces the fundamental architecture and methodologies in ECS-Grid. Section III introduces the simulator MessagePack protocol and the performance test of vIEDs with different transportations; the way to implement industrial protocols such as IEC-60870-5-104 is also discussed. Section IV presented the microgrid cluster study case with results from one steady-state scenario and one cyber-attack scenario. Section V is the conclusion.

II. Proposed Data-Oriented Architecture of ECS-Grid

A. Data-Oriented ECS Architecture

The information exchange between physical and cyber systems is the major concern in a cyber-physical simulation. Information is carried by data, and generally, cyber systems are built to transport and process data that carry useful information. However, while data can represent almost anything in the digital world, an object which contains both data structure and behaviors can only carry limited information whose pattern is pre-defined by its abstract templates: the *Class*, without the ability to mutate its structure. The *Inheritance* makes it even worse due to the extra dependencies between *Classes*. As shown in Fig. 1 (a), the OOP paradigm creates abstract base *Classes* for different domains while inherited implementations are realized in sub *Classes*. This adds difficulties in refactoring and optimization. Since cyber-physical systems in the big data age are transporting enormous unstructured data, a data-oriented solution that focuses on data processing and data combinations is highly preferred to OOP solutions.

Data-oriented programming means data combinations determine functionalities, which is also the core concept of ECS-Grid. The ECS framework starts to play a significant role in the game industry and modern software engineering, which is now the backbone of Minecraft™ [23], Data-Oriented Technology Stack (DOTS) in Unity® [24], Call of Duty®: Vanguard, ArcGIS Runtime SDKs by Esri™, and many modern large-scale commercial software projects. However, it is still not utilized for cyber-physical simulation in power industries which are full of data-intensive applications. Currently, there are three major types of ECS frameworks: bitset, archetype, and sparse-set, where sparse-set is the most popular one due to its high flexibility and archetype has the best theoretical performance. The specific types of components are managed by an entity registry to provide database-like access to the data objects. In this paper, the sparse-set-based EnTT [23] is used as the entity registry, which is also used in Minecraft™. Everything under the ECS framework belongs to an *Entity*, *Component*, or *System*. The inheritance is replaced by an entity's composition of data components in the ECS framework. An *Entity* is an integer identifier that is linked to multiple components in an entity data registry which can be seen as a data table in Fig. 1 (b). A *Component* is a structure with data to process. As shown in Fig. 1 (b), entities are rows of the data table. For example, a voltage-source converter (VSC) entity is composed of the four circles in a row of the table view; the EMT model object which is the same circuit object of the traditional OOP design such as an averaged-value model VSC; the IO module which holds measured signals and controller signals; the VSC controller holds data for control logics and the IED communication module holds the information of network sockets and other parameters such as latencies.

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2023.3244329

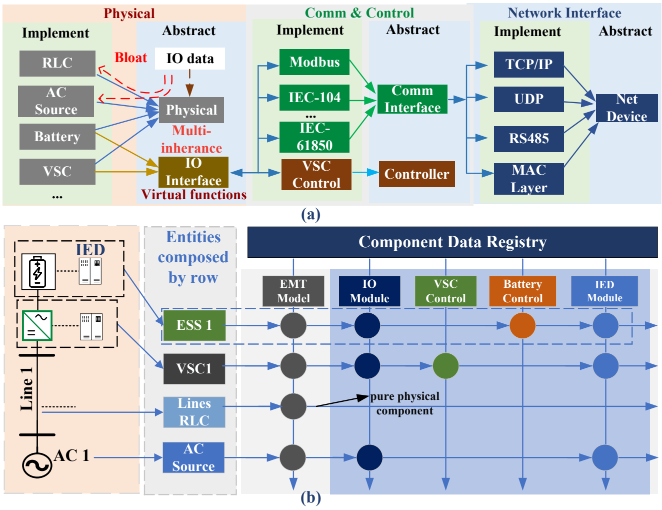


Fig. 1: Traditional OOP and data-oriented ECS design for CPPS simulation: (a) Inheritance and abstract interfaces create complex object relationships to represent physical objects with IED under the OOP paradigm. (b) An entity is defined by data component combinations similar to a row in a data table under the ECS framework, while data are processed by columns.

A specific combination of data components will be processed by relevant *Systems* which contain all program logic. A *System* is a plain function that can process columns of components in the table of Fig. 1 (b). It usually takes the registry as the input parameter and creates a query view of components from the registry just like a database query. The queries are optimized by the ECS framework depending on its storage type and usually are blazing fast. Because the system function is only called once on specific types of components, it eliminates the bloating issue caused by intermediate interfaces or CPU overhead in dynamic virtual function calls per object; since the components are stored in compact arrays, it is also more cache-friendly and easier to take advantages of modern single-instruction-multiple-data (SIMD) hardware such as graphical processing units (GPUs). In other words, it can fully avoid the usage of inheritance and polymorphism to build more complex and efficient software with an ECS framework. Also, ECS brings impressive flexibility to modify an *Entity*. For example, one can replace the VSC control component without breaking other VSCs with the same physical model, or replace a system at the runtime to change the functionality.

The proposed ECS-Grid currently uses a hybrid ECS solution to fully reuse the traditional OOP EMT simulation code. The EMT simulation loop is untouched, and no modification is added to any physical component class. The only difference is the traditional physical components are now managed as a part of an entity in an ECS registry instead of an all-in-one object. The IED feature is added by introducing new components and systems to the physical software. This hybrid solution can be very useful for industrial developers to transfer from traditional OOP to data-oriented design under the ECS framework. The full transition to an ECS data-oriented simulation framework requires many critical changes to traditional design patterns and still needs some exploration. Details about *EMTModel* are discussed in Section II.B. IED components and systems are discussed in Section II.C.

B. Physical EMT Simulation

Currently, only EMT physical simulation is implemented in ECS-Grid. In EMT simulation, dynamic physical components such as capacitors and inductors are represented by differential equations. Small and simple circuits such as RC, LC, and RLC can be solved with a form of ordinary differential equations or state-space format. However, a power grid usually consists of a large amount of different physical components, which is more suitable to solve as differential-algebraic equations (DAEs) with nodal analysis. The nodal analysis is based on solving a circuit equation system with nodal voltages as primary unknown variables. Previous research works often emphasize elementwise derivation, while in this paper the simulation is purely expressed by the language of linear algebra. Using Kirchhoff's Current Law (KCL) any RLC circuits with k nodes (ground node excluded) can be represented by:

$$\begin{aligned}\sum \boldsymbol{i}_{out} &= \boldsymbol{i}_L + \boldsymbol{i}_C + \boldsymbol{i}_G \\ &= W_L \int \boldsymbol{v} dt + W_C \frac{d\boldsymbol{v}}{dt} + W_G \boldsymbol{v} = \boldsymbol{s}, \\ W_L = B_L^T[\frac{1}{\boldsymbol{L}}]B_L, &W_C = B_C^T[\boldsymbol{C}]B_C, W_G = B_G^T[\boldsymbol{G}]B_G,\end{aligned} \tag{1}$$

where B is the oriented incidence matrix whose rows are corresponding to the physical components and columns are corresponding to nodes, L is the inductance, C is the capacitance and G is the admittance, $\boldsymbol{s}$ is the vector of current injections by sources. B is a transformation to gather the port voltages from global nodal voltages $\boldsymbol{v}$, while B^T can scatter the branch currents into the nodal injection vector. The W matrices are the weighted Laplacian matrices of different types of components, which are also called admittance matrices and play important roles in solving power grid equation systems. $[X]$ means diagonalized matrix of 1-D vector X.

To solve (1) with the Trapezoidal Rule, the following equations can be obtained:

$$\sum \boldsymbol{i}_{out_{n+1}} = \boldsymbol{i}_{L_{n+1}} + \boldsymbol{i}_{C_{n+1}} + \boldsymbol{i}_{G_{n+1}} = \boldsymbol{s}_{n+1}, \tag{2}$$

$$\begin{aligned}(W_G + \frac{2}{\Delta t}W_C + \frac{\Delta t}{2}W_L)\boldsymbol{v}_{n+1} &= \boldsymbol{s}_{n+1} + \boldsymbol{I}_{Leq}^{n+1} + \boldsymbol{I}_{Ceq}^{n+1}, \\ \boldsymbol{I}_{Leq}^{n+1} &= -\frac{\Delta t}{2}W_L\boldsymbol{v}_n - \boldsymbol{i}_{L_n}, \\ \boldsymbol{I}_{Ceq}^{n+1} &= \frac{\Delta t}{2}W_C\boldsymbol{v}_n + \boldsymbol{i}_{C_n},\end{aligned} \tag{3}$$

where $\boldsymbol{I}_{Leq}^{n+1}$ and $\boldsymbol{I}_{Ceq}^{n+1}$ are synthetic current sources created by discretization. A similar rule is applied to all other dynamic or time-varying physical components in the power system for EMT simulation, which gives the general form:

$$\begin{aligned}Y\boldsymbol{v}_{n+1} &= \boldsymbol{s}_{n+1} + \sum \boldsymbol{I}_{eq}^{n+1}, \\ Y &= \sum W_{diff},\end{aligned} \tag{4}$$

where Y indicates the final admittance matrix for the system solution, which can be inversed to solve the primary unknown variable v_{n+1}; I_{eq}^{n+1} indicates all equivalent current sources generated by physical components; W_{diff} is the admittance matrix derived from differential equations such as $\frac{\Delta t}{2}W_L$ for inductors.

For nonlinear components such as diodes that have nonlinear voltage-current characteristics. (4) is extended to:

$$\begin{aligned} \boldsymbol{i}_N + Y\boldsymbol{v}_{n+1} &= \boldsymbol{s}_{n+1} + \sum \boldsymbol{I}_{eq}^{n+1}, \\ \boldsymbol{i}_N &= B_N^T[f(B_N\boldsymbol{v}_{n+1})] \end{aligned} \tag{5}$$

where f is an elementwise nonlinear function of $B_N\boldsymbol{v}_{n+1}$. Newton's method is utilized to linearize the system and solve it, which converts Equation (5) into the following:

$$\begin{aligned} F(\boldsymbol{v}_{n+1}) &= \boldsymbol{i}_N + Y\boldsymbol{v}_{n+1} - (\boldsymbol{s}_{n+1} + \sum \boldsymbol{I}_{eq}^{n+1}) = 0, \\ F'(\boldsymbol{v}_{n+1}^m) &= \frac{\delta \boldsymbol{i}_N}{\delta \boldsymbol{v}}|_{\boldsymbol{v}_{n+1}^m} + Y = J^m, \\ \boldsymbol{v}_{n+1}^{m+1} &= \boldsymbol{v}_{n+1}^m - \frac{F(\boldsymbol{v}_{n+1}^m)}{F'(\boldsymbol{v}_{n+1}^m)} = \boldsymbol{v}_{n+1}^m - J^{m^{-1}}F(\boldsymbol{v}_{n+1}^m), \end{aligned} \tag{6}$$

where m denotes the iteration index of the Newton method, J^m is the system Jacobian matrix at mth iteration. Equation (6) can be reorganized into:

$$\begin{aligned} J^m\boldsymbol{v}_{n+1}^{m+1} &= (J^m - Y)\boldsymbol{v}_{n+1}^m - \boldsymbol{i}_N^m + (\boldsymbol{s}_{n+1} + \sum \boldsymbol{I}_{eq}^{n+1}) \\ &= W_{nl}^m\boldsymbol{v}_{n+1}^m - \boldsymbol{i}_N^m + (\boldsymbol{s}_{n+1} + \sum \boldsymbol{I}_{eq}^{n+1}), \end{aligned} \tag{7}$$

where

$$W_{nl}^m = \frac{\delta \boldsymbol{i}_N}{\delta \boldsymbol{v}}|_{\boldsymbol{v}_{n+1}^m} = B_N^T[\nabla f(B_N\boldsymbol{v}_{n+1}^m)]B_N. \tag{8}$$

Therefore, the nonlinear components have the harmonized format of admittance matrices and artificial current injections, which gives the following recursion formula:

$$\begin{aligned} J^m\boldsymbol{v}_{n+1}^{m+1} &= \boldsymbol{s}_{n+1} + \sum \boldsymbol{I}_{eq}^{n+1} + \sum \boldsymbol{I}_{nleq}^{m+1}, \\ \boldsymbol{I}_{nleq}^{m+1} &= W_{nl}^m\boldsymbol{v}_{n+1}^m - \boldsymbol{i}_N^m. \end{aligned} \tag{9}$$

The Jacobian matrix needs to be assembled and inverted serval times in each simulation time-step. Therefore, the f of many nonlinear components may be converted into piece-wise linear function or use $\boldsymbol{v}_n$ to approximate $\boldsymbol{v}_{n+1}$ to speed up the computation. (1-9) cover the fundamentals of the physical power system *EMTModel* components and systems in ECS-Grid.

C. Data-Oriented IED Simulation

In most CPPS simulations, there were only two layers: the physical layer and the cyber layer, which often ignored first-class citizens in real-world CPPSs: intelligent electronic devices (IEDs). The IEDs are generally protection, control, and monitoring devices in power grids, which are cornerstones for the modern power system automation and smart grid [25]. Therefore a comprehensive cyber-physical simulation should model these IEDs to be as realistic as possible.

As shown in Fig. 2, a commonly used IED in power systems is composed of multiple modules: *DSP controller*,

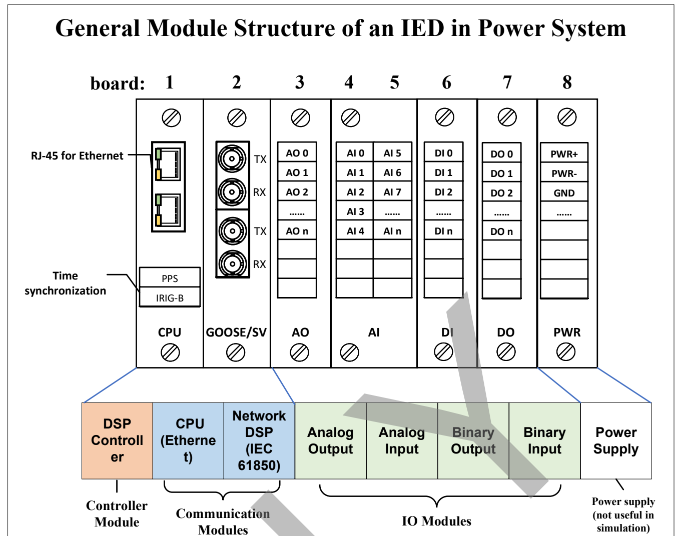


Fig. 2: A real-world IED consists of the controller, communications, IO modules, and power supply.

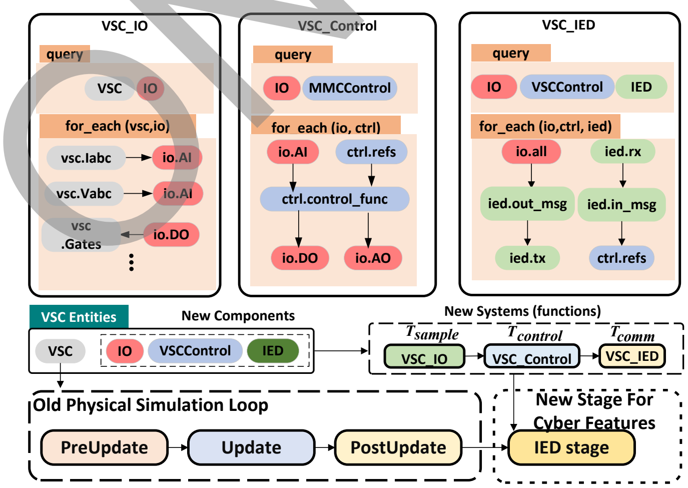


Fig. 3: IED sampling, control, and communication systems execution in the proposed data-oriented framework of ECS-Grid.

CPU, *Network DSP (only for optical IEC-61850 GOOSE/SV)*, *AI, AO, DI, DO* for analog or digital inputs/outputs (IOs), and the *power source*, which are circuit boards is responsible for specific tasks. Different configurations of the board modules and internal firmware will define the functionalities of the IED. It can be beneficial to model the IEDs inside EMT simulation to reflect real-world communication behaviors such as network latencies, time synchronizations, and protocol analysis, and also bring many new possibilities to cyber-physical research. The structure of real-world IEDs is a perfect match for the proposed data-oriented architecture.

ECS-Grid proposed the layer of vIEDs, which is an independent set of components and systems to model IEDs in power grids to conduct control or communication tasks. These digital twins of real-world IEDs bring more realistic and more consistent experiences from real-world CPPS. The vIEDs consist of IO, control, and communication components

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2023.3244329

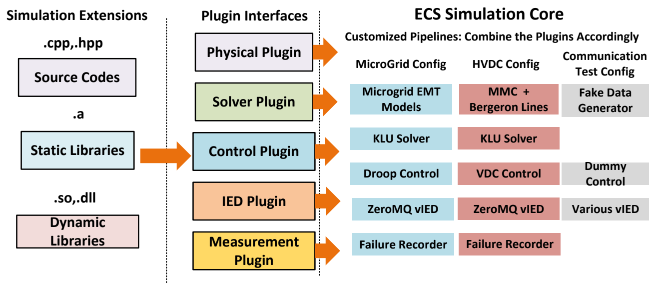


Fig. 4: Example plugins and their configurations in the proposed ECS-Grid.

in the ECS-Grid. which covers the fundamentals of a real-world IED in Fig. 2.

As shown in Fig. 3, the components of VSC entities in Fig. 3 are similar to the modular boards in Fig. 2; The systems:*VSC_IO*, *VSC_Control* and *VSC_IED* are similar to the software programmed into the physical IED; the systems are grouped and called in the *IED Stage* which is considered as a new sensing layer compared to old physical simulation loop. Similar to the replaceable boards and upgradable programs in real-world IEDs, the components and systems of vIEDs can be replaced or reorganized to serve different purposes both at compile-time and runtime. This is realized elegantly within a data-oriented ECS framework while OOP cannot compose it nicely due to its fixed pre-defined structures. The vIED extension is implemented by a very simple plugin interface introduced in Section II-D.

D. Plugins Made Easy

The extensibility is important for a cyber-physical simulation platform and that should be a significant advantage of a data-oriented design. The functionalities of ECS-Grid are defined by a combination of plugins, which is similar to many popular ECS frameworks such as Flecs and Bevy. Plugins can have various inner structures and definitions as long as they provide a plain function with a declaration of *void build(World &world);* as the entry point. In this way, a plugin with functionalities in Fig. 3 can be loaded from a header-only library, a static library, and even a dynamic library loaded at run-time. The implementations are quite straightforward and an example C++ header of the ZeroMQ vIED plugin is included in Appendix A.

As shown in Fig. 4, a simulation based on the ECS framework is composed of various plugins, which is flexible and bloat-free. For example, although the solver and vIED plugins are the same, the simulation for microgrids uses exclusive microgrid systems such as renewable energy sources and storage units along with the droop controllers, while the HVDC simulation configuration only uses the MMC and Bergeron line model plugins. To test the vIED only, the physical plugins are removed and replaced by dummy data sources. These configurations are practically applied to produce the results presented in Section III and IV.

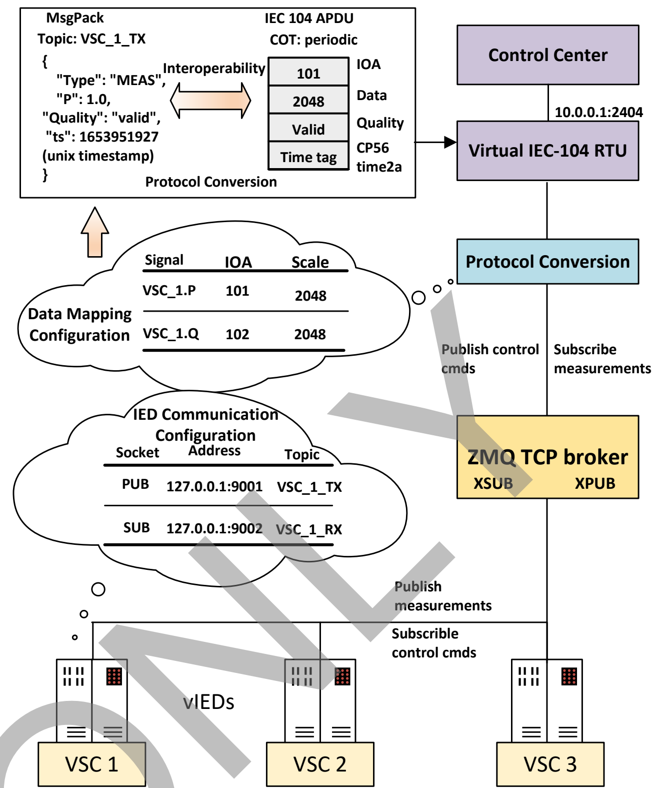


Fig. 5: The communication between vIEDs and real-world IEC 60870-5-104 clients.

III. Proposed Data-Oriented Protocol for Real-Time Cyber-Physical Simulation

Traditional CPPS simulations are usually based on available commercial simulators, where the signals of the physical power grid are grouped, converted to industrial protocols, and sent to cyber simulation machines. However, the industrial protocols are designed for production environments which should consider security issues, standard requirements, guidelines of power system operations, and the limits of existing industrial communication routes and devices. However, the simulation environments should provide a more generic protocol to simulate various scenarios which cannot be covered by a single industrial protocol. Also, the simulator itself should provide exclusive remote control and management functionalities for simulation-only purposes which are not considered by industrial protocols and IEDs.

Although some platforms [14] use Open Platform Communication (OPC) or CORBA (Common Object Request Broker Architecture) DIM (Distributed Information Management) protocol to unify the protocols within the simulator, these traditional OOP protocols are based on late-1990s standards and technologies which cannot meet the data-oriented demands of modern cyber-physical simulation. The OOP protocols often need a cumbersome object library to decode the messages and many functionalities are fixed. Therefore, a data-oriented protocol and a local simulation network are proposed for the vIEDs as a unified middleware interface to the outer systems. The data-oriented protocol should be:

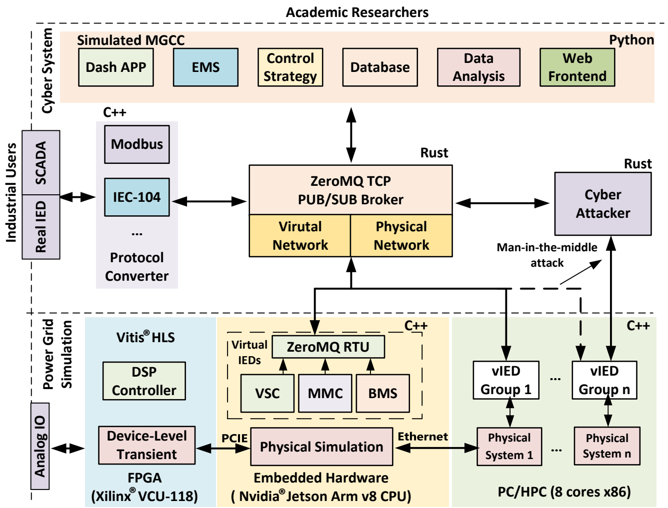


Fig. 6: The distributed network architecture of the ECS-Grid platform.

(1) generic: it should be able to represent different messaging patterns used in microgrids and not be restricted to specific transport media;

(2) high-performance: it should not add heavy overhead to the simulation systems and can handle a large volume of data; it should have distributed and concurrent features to make full use of modern hardware;

(3) customizable: unlike industrial protocols where all are defined by standards, the CPPS simulation should enable more possibilities for research explorations of future power systems by allowing users to customize the protocol.

A. MessagePack Format for User Applications

The ZeroMQ mainly abstracts the sockets for higher-level applications, the payloads being transported depends on the user's decision. The default vIED plugin uses a simple solution based on javaScript object notation (JSON) and MessagePack is proposed for a generic and customizable application-layer simulation protocol. JSON is the first-class data format inside ECS-Grid for configurations and data exchanging shown in Fig. 5. It is faster and smaller than the current XML format used in industrial applications [26]. The JSON format is self-describing, so there are no complex data models predefined by an Interface Definition Language (IDL) to decode the messages. MessagePack is an efficient binary serialization format and it can exchange JSON data cross multiple languages more efficiently [22]. The receivers can easily decode the messages to JSON objects like dictionaries in Python, ECMAScript, and Rust and handle them in their program logic. The utilization of MessagePack can provide a faster serialization and deserialization speed without a pre-defined schema and reduce the size by more than 40% compared to a plain-text JSON message. The example for the JSON-like protocol and the protocol conversion. The MessagePack design enables users to simulate specific scenarios and make custom virtual cyber services such as microgrid control center (MGCC) upon vIEDs, which can provide a very convenient platform for developing future distributed multi-layer control schemes and other cyber components as shown in Fig. 6.

Real-time communication performance is easy to achieve since simulation environments have much better computing power, bandwidth, and reliability than field devices. Modern CPUs have quite a large memory bandwidth that is larger than high-end optical networks. For example, Intel® Core™ X-Series Processors can achieve a bandwidth of 94GB/s with 4-channel DDR4 2933Mhz memories [27], which is nearly 8 times faster than high-end 100Gbit/s Ethernet. The current 10/100Mbps industrial Ethernet bandwidth is no match to the CPU's internal bandwidth.

Although the customizable protocols are useful for simulation environments, the industrial protocols cannot be ignored. As shown in Fig. 5, protocol converters are the solutions, which map the MessagePack protocol to a specific protocol such as IEC 60870-5-104. Protocol converters are common in real-world power automation systems and many IEDs can do protocol conversions internally according to the firmware or hardware configurations. Thanks to the multi-transportation ZeroMQ, the protocol conversion can have multiple choices to meet users' demands and interoperability can be ensured by customizing MessagePack messages. If users want industrial protocols directly built into the IED, they can follow the same plugin development principles to integrate their protocols.

B. Comparsion of Various Middleware Protocols

Currently, there are three communication plugins available for vIEDs in ECS-Grid: ZeroMQ [21], eProsima Fast DDS Real-Time Publish-Subscribe protocol (RTPS) [28], and Eclipse Paho MQTT [29]. Fast DDS is the middleware used in Robot Operation System 2 (ROS2). The MQTT is used for Internet-of-Thing (IoT) applications and partly in microgrid applications with IoT devices. ZeroMQ is a widely used message-oriented middleware. The latency test results of different protocols under the one-publisher-one-subscriber vIED scenario are listed in Table I-V.

Table I shows the results from the Fast DDS RTPS protocol. There is a spike in maximum latency when the message number increases, which is normally due to unreliable User Datagram Protocol (UDP) transportation. In summary, RTPS's performance is high and stable, and it has advanced features which can be very useful for vIED applications. However, it requires many dependencies, and the provided advanced features are not used in power systems. Moreover, it is not easy to use and the support documents should be greatly improved compared to other solutions.

Table II shows the results from Eclipse MQTT Paho clients. The MQTT is not designed for microsecond-level latency, and it must have a broker, which is an Eclipse Mosquitto broker [30]. The default configuration also enables message persistence on the broker server. Therefore, the latency is 1-100ms level which is good for most IoT applications but not good for low-latency communications. However, the bandwidth reaches the top of all protocols when the published message number is 1000. In all, the MQTT solution can be useful for some IoT scenarios since not all devices need microsecond-level latency.

Table III- V shows the ZeroMQ vIED performance under different configurations. The in-memory inter-thread commu-

TABLE I: vIED Latency and Bandwidth Using eProsima Fast DDS (RTPS)

Messages	Max (μs)	Min (μs)	Mean (μs)	Pub Bandwidth (Mbit/s)
100	98.70	14.42	16.99	34.28
1000	211.44	12.77	15.80	36.96
10000	293.81	13.11	14.77	38.15
1000000	7483.24	11.88	15.06	38.63

TABLE II: vIED Latency and Bandwidth Using Eclipse Paho MQTT

Messages	Max (ms)	Min (ms)	Mean (ms)	Pub Bandwidth (Mbit/s)
100	99.81	95.75	96.86	22.21
1000	99.87	75.60	92.67	71.41
10000	140.22	27.90	80.04	18.34
1000000	145.71	0.92	78.44	15.20

TABLE III: vIED Point-to-Point Latency and Bandwidth Using ZeroMQ (Inter-Thread)

Messages	Max (μs)	Min (μs)	Mean (μs)	Pub Bandwidth (Mbit/s)
100	203.21	6.63	25.23	18.28
1000	210.52	6.69	10.26	47.91
10000	191.43	6.37	7.88	60.40
1000000	230.29	6.05	7.08	67.03

TABLE IV: vIED Point-to-Point Latency and Bandwidth Using ZeroMQ (TCP)

Messages	Max (μs)	Min (μs)	Mean (μs)	Bandwidth (Mbit/s)
100	366.13	20.76	84.35	17.50
1000	396.42	16.44	24.87	48.28
10000	347.94	15.49	20.70	60.59
1000000	332.94	15.04	18.48	62.83

TABLE V: vIED Latency and Bandwidth Using ZeroMQ (TCP with Broker)

Messages	Max (μs)	Min (μs)	Mean (μs)	Pub Bandwidth (Mbit/s)
100	343.06	34.46	64.24	54.05
1000	266.82	34.81	52.09	61.29
10000	357.40	33.43	43.15	55.50
1000000	399.11	26.02	40.98	61.54

nication reaches the lowest latency of 6μs, which is quite enough for IEC-104 applications since the protocol timestamp has a resolution of milliseconds. The ZeroMQ point-to-point TCP pub/sub latency is around 20μs and the TCP pub/sub with a broker test is just a doubled point-to-point TCP latency. The single-thread publisher's bandwidth is high and stable without throughput optimization and well suited for a real-world IED which mainly has a 100Mbit/s Ethernet port. Besides, ZeroMQ is quite flexible and easy to use in every major programming language. The only problem is it requires more user decisions to establish an in-production network, however, it is an advantage for a simulator that can give users the maximum freedom to establish customized scenarios.

For the generic and high-performance goals, ZeroMQ is recommended to be the message bus between vIEDs. ZeroMQ is a high-performance asynchronous messaging library, aimed at use in distributed or concurrent applications. ZeroMQ supports scalability protocols (pub/sub, request/reply, client/server, and others) over a variety of transports (TCP, in-process, inter-process, multicast, WebSocket, and more). This keeps the code clear, modular, and scalabe from very low-latency in-memory communication to the large-scale cloud computing scenario.

IV. Case Study, Results and Performance

Fig. 7 shows the microgrid cluster connected by a multi-terminal DC system, which forms a 711-node power system with 60 vIEDs to evaluate the proposed simulation platform's functionalities and performance. The microgrid is a 15-Bus power distribution system derived from the CIGRE report [31] and pandapower [32] case files; loads are reduced to 10% and a 5MW Li-ion battery storage is added to Bus-1 to ensure the ability of islanded operation. The microgrid has 16 VSC stations and they are all modeled by the average-value model to reduce the complexities of control and simulation. Each VSC station has a VSC controller and a vIED acting as a remote terminal unit (RTU) to the VSC station. Each battery storage has an extra vIED to control the battery charging. The distributed power sources are controlled as PQ nodes which have fixed power generations, while the storage stations are controlled by a droop controller to auto-balance the system and provide a stable frequency. The loads are modeled by fixed RLC components for convenience. Three modified CIGRE 15-Bus microgrids are connected to the three-terminal high-voltage direct current (HVDC) system. The ±50kV HVDC system consists of three 51-level 50MW modular multilevel converters (MMCs) and MMC-1 is designated to control the DC voltages. The other 2 MMCs are set to drain 1MW from the HVDC system. The MMCs are modeled by detailed-switching models which means voltage balancing of submodules is needed. In this paper, the nearest-level modulation is used for MMC's lower-level controller. The upper-level controllers for MMCs are similar to VSCs in microgrids which control the DC voltages or the power generations.

Fig. 8 shows two scenarios for the test results. *Droop_0* IED in MG-1 is the main research target. **Scenario 1** is used to evaluate the islanded microgrid clusters and produce the steady state for **Scenario 2**. **Scenario 2** conducts a man-in-the-middle cyber attack to manipulate secondary frequency regulation command and cause catastrophe across the cluster. Scenarios 2 is similar to the real-world industroyer cyber attack in 2016 and industroyer2 attack in 2022 on Ukraine power grids [33], which hijacked supervisory control and data acquisition (SCADA) systems and sent dangerous commands to IEC-104 RTUs and IEDs.

Fig. 9 shows the setup of the real-time hardware platform introduced in Fig. 6. The three NVIDIA® Jetson AGX Xavier embedded computers with real-time Linux installed are used to simulate physical microgrids, the corresponding MMC station, and vIEDs. The Xilinx® VCU118 board is used to handle fast signal IO to support hardware-in-the-loop functions. The PC server runs cyber services such as virtual MGCC, IEC 60870-5-104 clients, and cyber simulation tools.

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2023.3244329

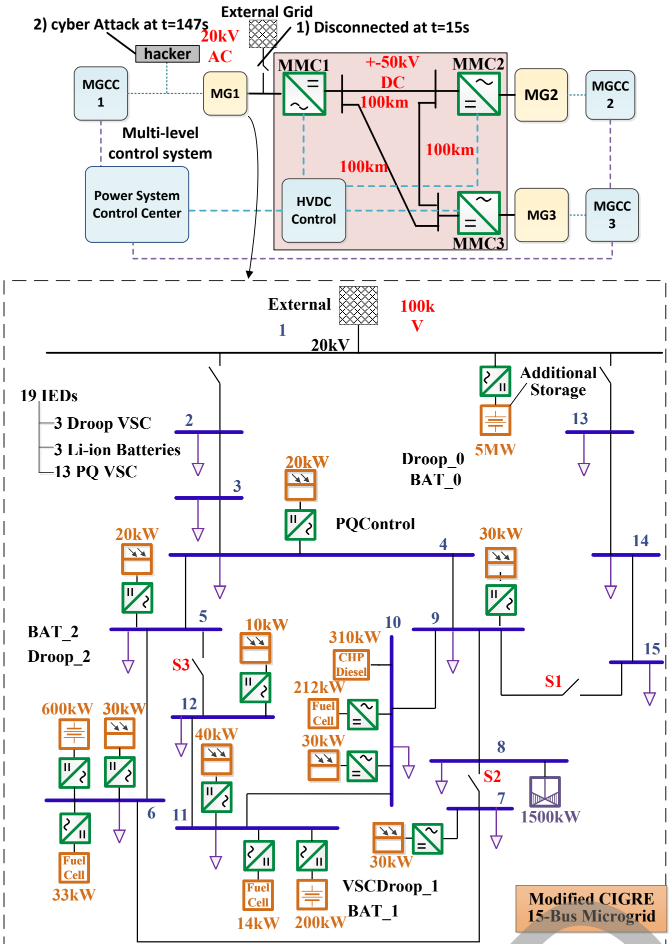


Fig. 7: System topology and the detailed configuration of the 711-node microgrid cluster.

A. Results and Performance

The results of **Scenario 1** are shown in Fig. 10 (a)-(f). Fig. 10 (a)-(c) are the frequency, real power, and bus voltage waveforms of *VSCDroop_0* storage station in each microgrid, respectively; when $t < 15s$, the frequency in MG-1 F_{Droop1} is 50Hz since the ideal three-phase AC source is attached to MG-1 Bus-0, while the other microgrids have different frequencies; at $t = 15s$ the ideal source is removed so that a large deviation occurred to F_{Droop1}, while MG-2 and MG-3 have no obvious changes because the DC system can allow asynchronous frequencies; the steady-state value of F_{Droop1} is 50.02 and P_{Droop1} is 4.0MW (0.8 p.u.), which meets the droop control equation $F - F_{ref} = K_p(P_{ref} - P) = 0.02$. This is considered a primary frequency regulation process. Fig. 10 (d)-(f) are the DC voltage, real power, and reactive waveforms of MMC stations, respectively; the rated DC voltage is 100kV so that all MMC stations maintained a nominal voltage according to Fig. 10 (d), while MMC-1 was impacted by the disconnecting event at $t = 15s$; Fig. 10 (e) shows the real power balancing between three MMCs, where MMC-1 provides 2MW and other MMCs drain the planned 1MW from the DC system; the reactive power was set to 0MVar, however, it seems to have large deviation at MMC-1 which may cause the larger

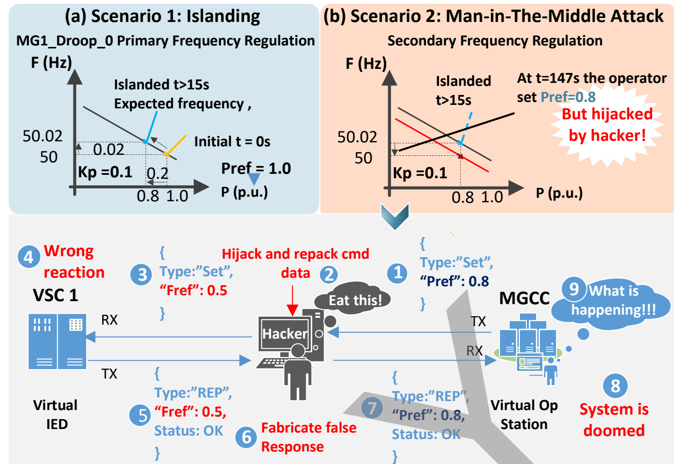


Fig. 8: Configurations and expected results of test scenarios: (a) **Scenario 1**: islanding operation; (b) **Scenario 2**: Man-in-the-middle cyber attack.

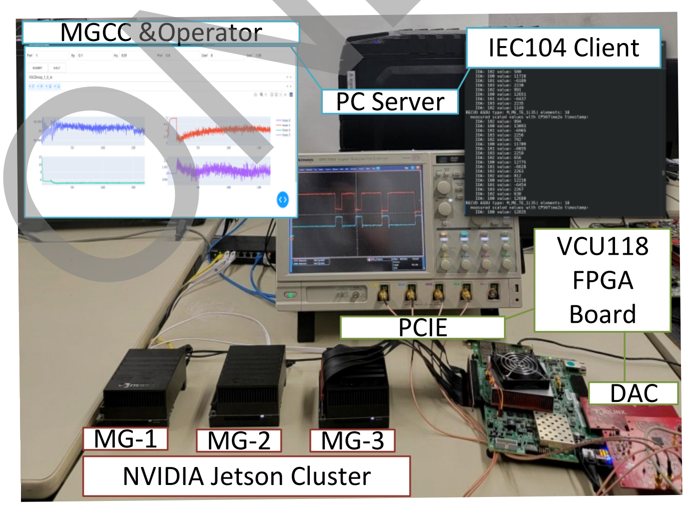


Fig. 9: Distributed hardware setup of proposed real-time ECS-Grid platform.

voltage fluctuation in MG-1. The results are verified against theoretical analysis and commercial PSCAD/EMTDC®; all data are measured from vIEDs with the interval of $100ms$ and recorded by remote self-made Supervisory Control and Data Acquisition (SCADA) system.

The results of **Scenario 2** are shown in Fig. 10 (g)-(i).

TABLE VI: FTRT Performance with Various Communication Intervals

Interval	T_{com}	t_{cps} (s)	r_{ftrt}	Efficiency
2	$100\mu s$	17.88	2.24	0.44
20	1ms	11.02	3.63	0.71
200	10ms	10.24	3.91	0.77
1000	50ms	8.08	4.95	0.97
2000	100ms	8.07	4.96	0.98

System Scale: 711 nodes, 60 vIEDs in total; Simulation duration: $40s$, $\Delta t = 50\mu s$, $t_{phy} = 7.87s$.

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2023.3244329

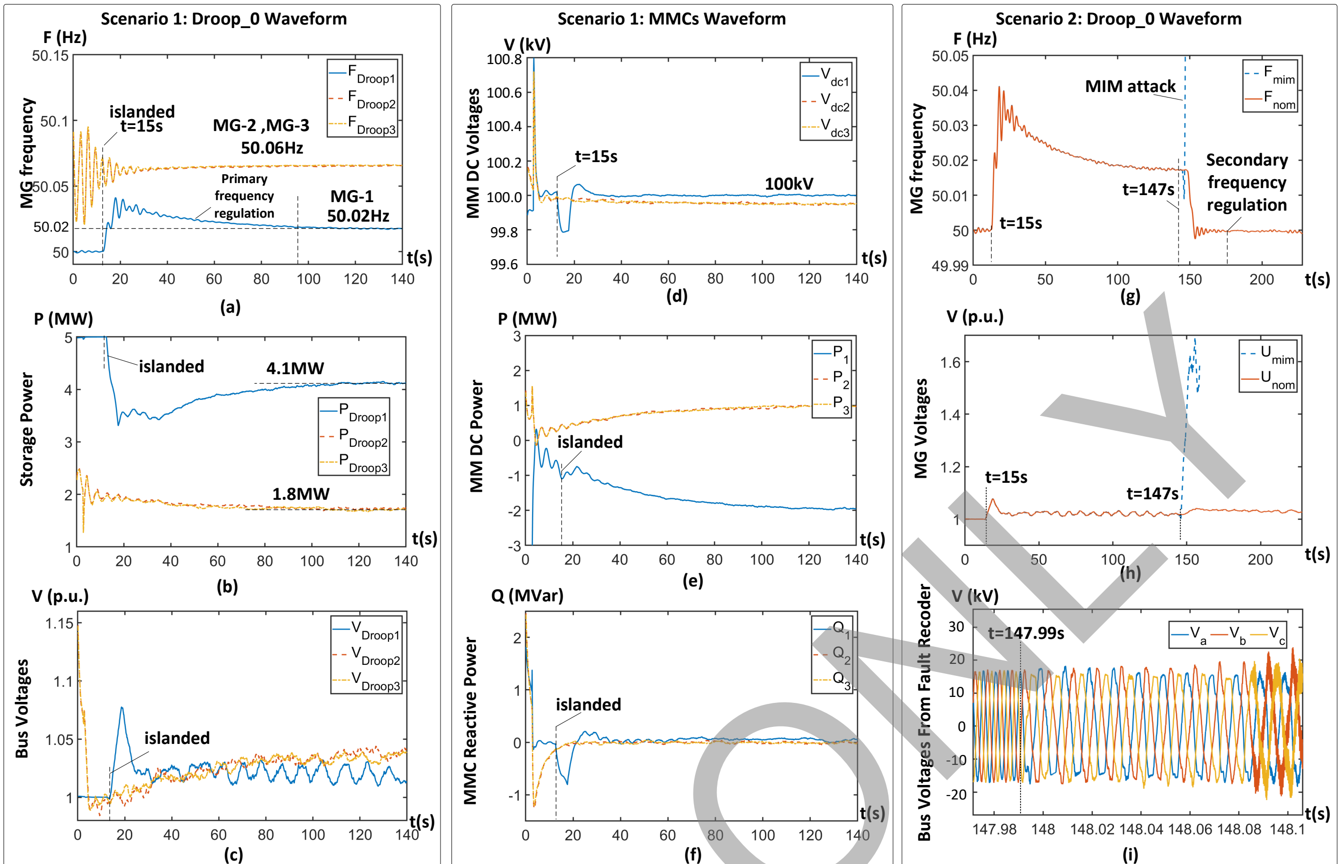


Fig. 10: Simulation results: **Scenario 1: islanding operation**:(a) Frequency of three *Droop_0* stations in each microgrid . (b) Real power of each *Droop_0* station. (c) Bus Voltages of each *Droop_0* station. (d) MMC DC voltages during the islanding operation. (e) Real power of each MMC station. (f) Reactive Power of each MMC station. **Scenario 2: simulated cyber attack**: (g) Frequency comparison between normal operation and cyber attack situation of *Droop_0* in MG-1. (h) *Droop_0* Bus voltages comparison. (i) *Droop_0* EMT bus voltage waveforms captured by virtual fault recorder.

the dashed line indicates the waveforms under cyber attack while the solid line indicates the normal reactions; since the frequency of MG-1 was 50.02Hz after islanded, the operator sent a command at $t = 147s$ to reset P_{ref} to 0.8 since the current real power is 0.8 p.u., which is a secondary frequency regulation process to restore rated operation point. Fig. 10 (g) shows the frequency of *VSCDroop_0* in MG-1; under cyber attack situation, the frequency regulation command was intercepted and replaced to $F_{ref} = 0.5p.u.$, which generates very drastic deviations in all measurements from IEDs after $t = 148s$ such as the voltages in Fig. 10 (h). However, these drastic deviations are measured from PLLs and may not reflect the real situation in the physical systems under faulty conditions and that is why real-world power systems also have digital fault recorders to record the EMT waveforms when faults occurred. Fig. 10 (i) shows the EMT voltage waveforms of Bus-1 captured by virtual digital fault recorders, which are triggered by the fault detection mechanism in the proposed simulation platform. The EMT waveforms revealed the physical details that happened after $t = 147s$; the system started to react about $1s$ later than receiving the hacked message, and the drastic deviations in measurements may be caused by the high-frequency oscillation which can affect stability. The simulation of Scenario 2 shows the catastrophic consequence of cyber attacks in a vulnerable power cyber network.

Besides the real-time Jetson platform, Table VI shows the performance of the proposed cyber-physical simulation platform on an x86 machine (Intel® Core™ i7 10700k 8c16t@4.7GHz, 32GB DDR4 3000MHz, Ubuntu 20.04, GCC 11.1). The pure physical parallel simulation consumes 7.87s which is 5.08 times faster than real-time, and all cyber-physical simulations also achieved faster-than-real-time (FTRT) performance even with $100\mu s$ ticking interval. For the millisecond-level communication intervals, the cyber-physical co-simulation can achieve high efficiency and the overhead is almost deflectable. The overhead can be further reduced with more concurrency for socket polling, data encoding, and decoding since the communication systems currently execute in series inside the main simulation loop. The FTRT functionality can enable predictive and preventative control actions in energy control centers.

V. Conclusion

ECS-Grid is a novel data-oriented cyber-physical simulation platform for microgrids under the ECS framework proposed to model the IEDs in a power system with flexible data components and extensible plugin architecture. Furthermore, a modern JSON-like MessagePack-based protocol is proposed for the vIEDs and is capable of completing various tasks needed for cyber-physical transient simulation. The results from the scenarios in the microgrid cluster study case show the accurate system behaviors and real-time or FTRT performance of ECS-Grid. The IED systems and components can be extended to cyber-physical power dynamic or steady-state simulations thanks to the data-oriented design. The data-oriented ECS-Grid can inspire the renovation of industrial software tools and boost further research of the future CPPS.

Appendix A

ZeroMQ C++ Plugin Header for ECS-Grid

```
#pragma once
// details of components
#include "components.hpp"
#include "entt/entity/fwd.hpp"
// details of systems
#include "vIED/zmq/IEDSystems.hpp"

// ECS container
#include "world/world.hpp"

/// This is your module namespace
namespace vIED
{
namespace zmq
{
/// declare your components
namespace Components
{
struct ZMQIEDContext;
struct IEDMessageBuffer;
struct IED;
struct IEDSocket;

/// a set of IED components
/// can be grouped into a bundle
using IEDBundle = basic::Bundle<IED,
IEDMessageBuffer, IEDSocket, Name>;

}  // namespace Components

/// declare your customized stage
enum class vIEDStage
{
  Communicate

};

/// declare your systems
namespace vIEDSystem
{
/// this initialize context and sockets
void initialize(entt::registry& reg);
/// this send messages using tx socket
/// including serialization
void sender(entt::registry& reg);
/// this receive messages using rx socket
/// including deserialization
void receiver(entt::registry& reg);
};  // namespace vIEDSystem

struct Plugin
{
  static void build(ecs::World& world)
  {
    // register your systems to the simulator
    using namespace vIEDSystem;

    world.add_startup_system_to_stage(PostStartUp,
                                      initialize);
    auto&& stage =
    world.add_stage_after(PostUpdate,
                          vIEDStage::Communicate);

    world.add_system_to_stage(stage->name(), sender);
    world.add_system_to_stage(stage->name(), receiver);
  }
};
}  // namespace zmq
}  // namespace vIED
```

References

[1] H. J. Schellnhuber, S. Rahmstorf, and R. Winkelmann, “Why the right climate target was agreed in paris,” *Nature Climate Change*, vol. 6, no. 7, pp. 649–653, 2016.

[2] S. Bolognani, R. Carli, G. Cavraro *et al.*, “On the need for communication for voltage regulation of power distribution grids,” *IEEE Trans. Control. Netw. Syst.*, vol. 6, no. 3, pp. 1111–1123, Sept. 2019.

[3] N. Patari, V. Venkataramanan, A. Srivastava *et al.*, “Distributed optimization in distribution systems: Use cases, limitations, and research needs,” *IEEE Trans. Power Syst.*, p. [Early Access], 2021.

[4] D. K. Molzahn, F. Dörfler, H. Sandberg *et al.*, “A survey of distributed optimization and control algorithms for electric power systems,” *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2941–2962, Nov. 2017.

[5] D. Gutierrez-Rojas, P. H. J. Nardelli, G. Mendes *et al.*, “Review of the state of the art on adaptive protection for microgrids based on communications,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 3, pp. 1539–1552, 2021.

[6] I. Ali and S. M. Suhail Hussain, “Communication design for energy management automation in microgrid,” *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 2055–2064, May 2018.

[7] R. V. Yohanandhan, R. M. Elavarasan, P. Manoharan *et al.*, “Cyber-physical power system (CPPS): A review on modeling, simulation, and analysis with cyber security applications,” *IEEE Access*, vol. 8, pp. 151 019–151 064, 2020.

[8] K. Mets, J. A. Ojea, and C. Develder, “Combining power and communication network simulation for cost-effective smart grid analysis,” *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1771–1796, 2014.

[9] IEEE Task Force on Interfacing Techniques for Simulation Tools, S. C. Müller, H. Georg *et al.*, “Interfacing power system and ICT simulators: Challenges, state-of-the-art, and case studies,” *IEEE Trans. Smart Grid*, vol. 9, no. 1, pp. 14–24, Jan. 2018.

[10] A. A. Jahromi, A. Kemmeugne, D. Kundur *et al.*, “Cyber-physical attacks targeting communication-assisted protection schemes,” *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 440–450, Jan. 2020.

[11] K. Hopkinson, X. Wang, R. Giovanini *et al.*, “EPOCHS: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components,” *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 548–558, May 2006.

[12] H. Lin, S. S. Veda, S. S. Shukla *et al.*, “GECO: Global event-driven co-simulation framework for interconnected power system and communication network,” *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1444–1456, 2012.

[13] H. Georg, S. C. Müller, C. Rehtanz *et al.*, “Analyzing cyber-physical energy systems:the INSPIRE cosimulation of power and ICT systems using hla,” *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2364–2373, 2014.

[14] M. H. Cintuglu, O. A. Mohammed, K. Akkaya *et al.*, “A survey on smart grid cyber-physical system testbeds,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 446–464, Jan. 2017.

[15] E. Moradi-Pari, N. Nasiriani, Y. P. Fallah *et al.*, “Design, modeling, and simulation of on-demand communication mechanisms for cyber-physical energy systems,” *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2330–2339, 2014.

[16] M. Thornton, M. Motalleb, H. Smidt *et al.*, “Internet-of-things hardware-in-the-loop simulation architecture for providing frequency regulation with demand response,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 5020–5028, 2018.

[17] B. Chen, K. L. Butler-Purry, A. Goulart *et al.*, “Implementing a real-time cyber-physical system test bed in RTDS and OPNET,” in *2014 North American Power Symposium (NAPS)*, November 2014, pp. 1–6.

[18] M. Ni, H. Tong, L. Zhao *et al.*, “A flexible hardware-in-the-loop testbed for cyber physical power system simulation,” *IET Cyber-Phys. Syst. Theory Appl.*, vol. 4, July 2019.

[19] H. Tong, M. Ni, L. Zhao *et al.*, “A flexible hardware-in-the-loop testbed for cyber physical power system simulation,” *IET Cyber-Phys. Syst., Theory Appl.*, vol. 4, 12 2019.

This article has been accepted for publication in IEEE Transactions on Industrial Informatics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TII.2023.3244329

[20] G. Cao, W. Gu, C. Gu *et al.*, "Real-time cyber physical system co-simulation testbed for microgrids control," *IET Cyber-Phys. Syst. Theory Appl.*, vol. 4, February 2019.

[21] "An open-source universal messaging library," The ZeroMQ authors, July 22, 2022. [Online]. Available: https://zeromq.org/

[22] "messagepack," GitHub, July 22, 2022. [Online]. Available: https://github.com/msgpack/msgpack

[23] "Entt: Gaming meets modern c++," GitHub, July 22, 2022. [Online]. Available: https://github.com/skypjack/entt

[24] "Data-oriented technology stack (DOTS)," Unity, July 22, 2022. [Online]. Available: https://unity.com/dots

[25] I. Jahn, F. Hohn, G. Chaffey *et al.*, "An open-source protection ied for research and education in multiterminal hvdc grids," *IEEE Trans. Power Syst.*, vol. 35, no. 4, pp. 2949–2958, 2020.

[26] "FlatBuffer serialization C++ benchmarks," Google FlatBuffer, Nov. 12, 2022. [Online]. Available: https://google.github.io/flatbuffers/flatbuffers_benchmarks.html

[27] "Theoretical maximum memory bandwidth for intel® core™ x-series processors," Intel, July 12, 2021. [Online]. Available: https://www.intel.com/content/www/us/en/support/articles/000056722/processors/intel-core-processors.html

[28] "eProsima Fast DDS," eProsima, Nov. 12, 2022. [Online]. Available: https://www.eprosima.com/index.php/products-all/eprosima-fast-dds

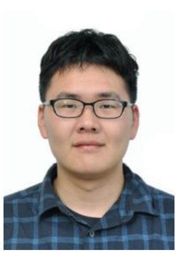
[29] "Paho is an IoT project," Eclipse Foundation, Nov 22, 2022. [Online]. Available: https://www.eclipse.org/paho/

[30] "Eclipse Mosquitto™," Eclipse Foundation, Nov. 12, 2022. [Online]. Available: https://mosquitto.org/

[31] C. Marnay, C. Abbey, G. Joos *et al.*, "Microgrids 1 engineering, economics, & experience," *Electra; CIGRE*, 2015.

[32] L. Thurner, A. Scheidler, F. Schäfer *et al.*, "pandapower - an open source Python tool for convenient modeling, analysis and optimization of electric power systems," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6510–6521, Nov. 2018.

[33] "Industroyer2: Industroyer reloaded," welivesecurity, Nov. 22, 2022. [Online]. Available: https://www.welivesecurity.com/2022/04/12/industroyer2-industroyer-reloaded/



Tianshi Cheng (S'19) received the B.Eng. degree in electrical engineering and automation from Southeast University, China, in 2017. From 2017 to 2018, he was a substation automation engineer of NARI Group Corporation (State Grid Electric Power Research Institute), China. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of Alberta, Canada. His research interests include electromagnetic transient simulation, transient stability analysis, real-time simulation, parallel processing, microgrid and power electronics.

Tong Duan (S'18-M'22) received the B.Eng. degree in electronic engineering from Tsinghua University, Beijing, China, in 2013, and the Ph.D. degree in energy systems from University of Alberta, Edmonton, Alberta, Canada, in 2021. He is currently an assistant professor at National Digital Switching System Engineering & Technological Research Center, Zhengzhou, China. His research interests include smart grid, industrial network, power system simulation and communication networking.

Venkata Dinavahi (Fellow, IEEE) received the B.Eng. degree in electrical engineering from Visvesvaraya National Institute of Technology (VNIT), Nagpur, India, in 1993, the M.Tech. degree in electrical engineering from the Indian Institute of Technology (IIT) Kanpur, India, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Ontario, Canada, in 2000. He is currently a Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada. He is a Fellow of the Engineering Institute of Canada. His research interests include real-time simulation of power systems and power electronic systems, electromagnetic transients, device-level modeling, large-scale systems, and parallel and distributed computing.