Article

S3DK: An Open Source Toolkit for Prototyping Synchrophasor Applications

Maxime Baudette [1,†,‡,*], Luigi Vanfretti [2,‡] and Shashank Tyagi [3]

1 Lawrence Berkeley National Laboratory, Berkeley, CA; baudette@lbl.gov
2 Rensselaer Polytechnic Institute, Troy, NY; vanfrl@rpi.edu
3 EirGrid Group, Dublin, Ireland
* Correspondence: baudette@lbl.gov
† Current address: 1 Cyclotron Road, Berkeley, CA, 94720
‡ These authors contributed equally to this work.

Abstract: Synchrophasor data contains a trove of information on the power system and its dynamics. These measurements have a high potential to unlock our ability to cope with changing system conditions and challenges posed by distributed and intermittent energy sources. While Phasor Measurement Units (PMUs) have seen a large deployment in the grid, their applications are limited by the software platforms that are deployed in control centers to monitor the grid. In this paper, we present an open source toolkit that enables fast prototyping of PMU applications. The toolkit is akin to a software development kit (SDK) for synchrophasor applications, providing a number of functionalities that enable high-level PMU application development within the LabVIEW environment. This Smart-grid Synchrophasor SDK (S3DK) proposes a paradigm based on the concept of distributed applications, which allows development and deployment to be independent from the existing software stack deployed in control centers and to leverage PMU data at any level of a synchrophasor system hierarchy. This paper serves to introduce the S3DK, which is released as open source software to facilitate broader and fast prototyping of synchrophasor applications.

Keywords: Power Systems; PMU; Synchrophasor; WAMS; Open Source

Citation: Baudette, M.; Vanfretti, L.; Tyagi, S. S3DK: An Open Source Toolkit for Prototyping Synchrophasor Applications. *Electronics* **2024**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

1. Introduction

The power grid is rapidly evolving with the addition of new technologies that relate to the many facets of the grid operation. One notable development is the broad deployment of measurement equipment, namely Phasor Measurement Units (PMUs). These units are characterized primarily by their ability to synchronously "measure" synchrophasors: a phasor representation of the three-phase system in the form of voltage and current (both magnitude and angle estimates). The deployment of the technology was originally motivated by a need for higher-fidelity measurements than those available from conventional Supervisory control and data acquisition (SCADA) systems, with the goal to capture evolving grid dynamics across wide geographical regions [1], an effort that gained impetus in the aftermath of the Northeast blackout of 2003 [2,3]. The time-synchronized and high-sample rate data was also as a means to facilitate more advanced computations to support situational awareness through wide-area monitoring [4], so to accommodate the operation of the grid with thinner security margins[5]. This is becoming especially relevant for the changes that have happened in the grid in the past decades, such as the expansion of synchronously interconnected grids, and the influx of inverter-based intermittent generation (e.g., renewables such as wind [6,7] and solar photovoltaic systems [8]).

Prior to the deployment of synchrophasor systems, grid operators relied primarily on SCADA systems, often coupled with a state estimation algorithm, to monitor and operate the grid. These tools do not allow for advanced monitoring functionalities as the measurements are collected in an asynchronous process, and only provide RMS values (i.e., amplitude) of the measured quantities. Synchrophasors, on the other hand, provide much richer measurements that enable the use of signal processing methods to extract more information, and combine measurements from several locations for more complex computations such as those in [9–12].

1.1. Motivation

The deployment of this technology stimulated the interest of both grid operators and researchers to develop new algorithms and tools that harness these measurements into actionable information for grid operation. The development of such applications at early stages is fundamentally related to the ability of the research community to develop and prototype their ideas. In the case of the power engineering community, its members do not necessarily possess the skill-set for advanced software development. This highlights the need for broadly available tools that abstract the complexity related to developing software and handling real-time measurements, such that power engineering students and researchers can access a user-friendly development environment to prototype applications with minimal training effort.

Commercial tools that support synchrophasor measurements are based on the same integrated and centralized architecture as traditional SCADA systems [13–16]. Hence, the development of new functionalities can only happen in close cooperation with their respective vendors, primarily through procurement processes, incurring high integration costs[1]. This is not an attractive alternative for fundamental research or prototyping purposes, which led to the development of various initiatives. There are a few notable contributions to the open-source synchrophasor applications community readily available [20], but the majority lack a user-friendly development environment for advanced algorithms and interactive user interfaces,hence, requiring advanced software development skills, even for the implementation of early prototypes.

This motivated the development of a new SDK that focuses on the specific needs of the power engineering community by providing access to PMU measurements in a user-friendly development environment; enabling power engineers to develop and prototype custom PMU applications.

1.2. Previous Work and Contribution

The release of this SDK is the result of a several years long effort that started with the development of a data mediator for the IEEE C37.118.2 standard protocol [21] for synchrophasor networks communications. This data mediator, the STRONg^2*rid* library [22], was developed to expose real-time PMU measurements through an Application Programming Interface (API) that could be further integrated into other applications. In particular, the API was populated with methods dedicated to its integration with LabVIEW in preparation of the development of the S3DK.

In this paper we present the Smart-grid Synchrophasor SDK (S3DK): an open source LabVIEW extension that enables the development of PMU applications. It provides the facilities to handle real-time PMU data, while the configuration is carried out through a Graphical User Interfaces (GUIs), and allows to leverage LabVIEW and its numerous toolboxes for signal processing, data analysis, and graphical outputs. The source code and the compiled installation files were released as an open-source project to further facilitate its adoption by researchers. Finally, to show how the S3DK can be leveraged by members of the power engineering community, we also compiled a list of the prototype tools that were developed by the research team that developed the S3DK, as well as some prototype tools that were developed by Grid Controller of India Limited, a utility company in India, and other third parties.

[1] See for example the case of the Grid Stability Awareness System (GSAS) [16] which integrated stability monitoring functions into OpenPDC with a cost of USD 3 Million [17], and other comparable cases [18,19].

1.3. Paper Organization

This paper is organized as follows: in Section 2, we cover synchrophasor systems and different paradigms for developing PMU applications Section 3 presents the S3DK and its main components and functionalities. In Section 4, we introduce the PMU application template that was integrated in the library package to fast track the prototyping of applications and summarize several PMU application prototypes developed using the toolkit. In addition, prototypes of applications developed by a utility company are presented in Section 5 Finally, conclusion are drawn in Section 6.

2. Synchrophasor Systems and Applications

Through different grid modernization efforts, several grid operators have deployed PMUs in their system to build a new monitoring system based on synchrophasors [23]. These devices sample all three phases of the grid at a fixed high-sampling rate to estimate a phasor representation of the phase voltages and currents at the fundamental frequency thereby providing both magnitude and angle "measurements"). The phasor estimation is performed synchronously across all devices using a common time reference (e.g., GPS) that also provides the reference to compute phase angles, hence the name synchrophasor. This particularity allows to use measurements from across the entire system and compute phase angle differences that are alike to those in power flow computations (i.e., line power flow equations).

2.1. Synchrophasor Systems

Traditional SCADA systems have a fully centralized architecture that is most often delivered by a single provider. The central server asynchronously collects all the measurements and present them in a single interface, and allow the operator to execute different monitoring and control actions. However, because the measurements in SCADA systems are polled asynchronously they need to be assisted by a state estimator to create a "snapshot" of observed and non-observable physical quantities (e.g., voltages, powers) in the grid.

Synchrophasor systems differ from the SCADA architecture in multiple aspects. Most importantly, synchrophasor systems stream data from PMUs, and the data flows are organized in a hierarchical system structure based on standardized communication protocols, such as the IEEE C37.118. At the core of the system, the Phasor Data Concentrator (PDC), is the component that collects and redistributes PMU measurements implementing the communication protocol. PDCs are typically used to interface different actors of the grid, such as private plant owners, regional grid operators, and the central system operator (Independent System Operators (ISO), Transmission System Operators (TSO), etc.). Figure 1 presents a typical hierarchical structure of the measurement data collection system of synchrophasor systems.

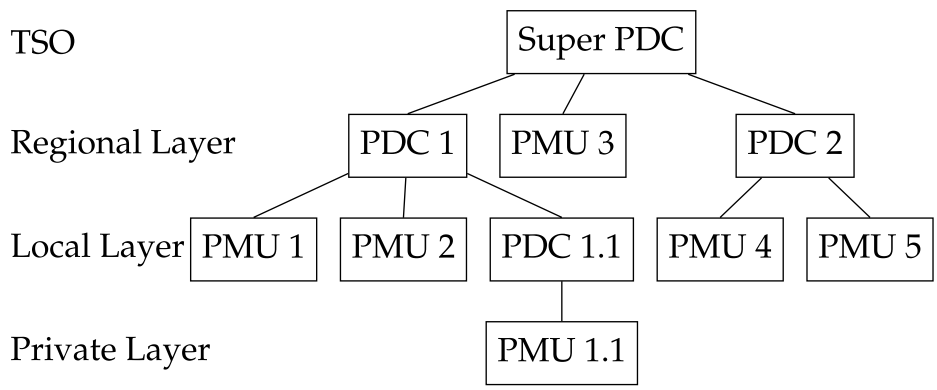

Figure 1. Synchrophasor System Architecture

PDCs support multiple outgoing connections, known as output streams, that allow to make use of the measurements at the different levels where they are placed for various kind of applications. There are two models for the application layer, i.e., where PMU applications are placed, that will be discussed in the following Section.

2.2. Applications Development Frameworks

The development of synchrophasor networks quickly stimulated the interest for finding new PMU data applications [3]. While early development focused on data collection and archival aspects for postmortem analysis of system disturbance event [2], the focus has expanded to the development real-time applications that help the decision making for grid operators [4], largely thanks to the advent of more processing power and higher throughput communication networks. This resulted in the development of different frameworks for developing synchrophasor applications that can either be part of the PDC platform [4,16,20], or less commonly, take the form of distributed application systems [24].

2.2.1. Centralized PDC-based Application Development

The prevailing centralized architecture is based on applications built in the PDC component and resembles the infrastructure of existing Energy Management Services (EMS) / SCADA systems. In this model, the PDC functionalities are extended through built-in applications (e.g., plugins) that can take advantage of private Application Programming Interfaces (APIs) offered by the PDC platform [25]. This approach allows for an unique and consistent interface with the PDC that may allow faster and more granular access to the measurement data, but comes at the cost of limiting the applications' scope to what the PDC framework can support (e.g., limited signal processing functions, etc.) without additional developments into the platforms with substantial costs [16–19].

For real-time application deployment, practically, this approach involves a single location for all computation requirements, co-located with the super PDC or PDC [16]. Thus, it has numerous implications for the development of applications by third parties contributor. While computation limitations of running concurrent applications on a single machine are no longer a source of concern, more difficulties can be expected for third parties to develop applications on often insufficiently documented platforms or those with a prohibitive cost to access. In particular, in an industrial setting, it is common in the power industry for ISOs/TSOs to rely solely on a few external vendors to develop and deploy their synchrophasor system, foregoing the possibility of broader third party contributions and delays the deployment of applications for real-world utilization.

2.2.2. Distributed Application Development and Deployment

The distributed application development approach differs from its centralized counterpart by relying on a real-time data mediator that consumes a PDC stream and deliver its content in a development environment. This also means that the application does not need to reside in a single location, and multiple instances of real-time data mediators can coexist and connect to any of the existing PMU/PDC system, as illustrated on Fig. 2. The real-time data mediator is implemented to parse a standard communication protocol (e.g., IEEE C37.118) into native datatypes of the chosen development environment [22]. The standardized protocol ensures the compatibility with any of the existing PMU/PDC systems.

This approach has many advantages for each party involved in the development and management of a synchrophasor system. From the system management perspective, it allows to focus the functionality of the synchrophasor system, and in particular the PDC, to data collection, management and redistribution. For synchrophasor applications it allows the deployment of a very modular infrastructure, which facilitates the deployment of more computational power, and new functionalities (i.e., new applications). This may also facilitate the creation of a very redundant system.

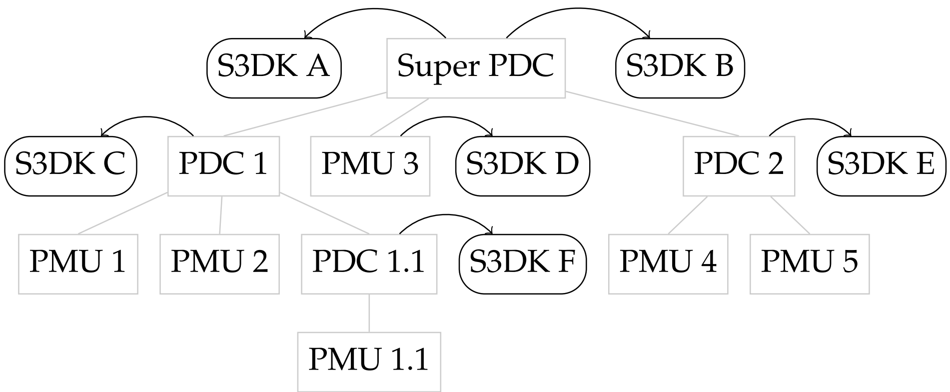

Figure 2. Example of Distributed Application Architecture in Synchrophasor System

From the application development perspective, it allows the use of broadly available development environments that can feature very extensive computational and, in particular, signal processing capabilities. It also opens the development of new applications to any academics, researchers and software developer who no longer needs to specialize in custom proprietary software environments. Finally it facilitates the transition from a development / testing environment, to a deployment in the field, by allowing easily packaged applications.

2.2.3. Existing Frameworks

In this paper, we focus on open source frameworks that can be used for developing synchrophasor applications. There are a few notable contributions to the open-source community that are readily available, including the following :

- RIAPS [24] is a framework for developing distributed monitoring and control applications,
- GridAPPS [26] is a framework for developing Advanced Distribution Management Systems (ADMS) applications that combine both model simulations and measurements,
- The OpenPDC project [27] is an open-source PDC that can be extended with custom *Adapters* developed in C# to extend its functionalities and computations,
- Synchro-Measurement Application Development Framework (SADF) [28] is a framework to develop synchrophasor applications in MATLAB.

RIAPS and GridAPPS offer a generic framework for grid applications that feature a communication facility. While these solutions do not explicitly support standardized synchrophasor streams [21], they provide tools that would allow such applications. OpenPDC offers a framework to develop applications on the centralized PDC based model. While it offers the possibility to develop new application using C#, this programming language is not broadly adopted in the power engineering scientific community and lacks toolboxes/libraries for advanced computations (e.g., signal processing, filtering, etc.). SADF offers a real-time data mediator for MATLAB. This programming environment is widely distributed in the scientific community, most engineers and researchers are acquainted to it, and offers a vast ecosystem of toolboxes for advanced computation. However, while SADF is open source software, it depends on MATLAB that requires costly licenses that becomes prohibitive when using multiple of its toolboxes. Nevertheless, among these frameworks, SADF is the closest solution to the S3DK presented in this paper.

3. S3DK: Smart-grid Synchrophasor SDK

The original concept of the Smart-grid Synchrophasor Software Development Kit was instigated to answer the needs of the research group led by the second author, to develop and test prototypes of various PMU applications. At the time (circa 2011) there was no readily available solution that allowed power system researchers, who often lack a

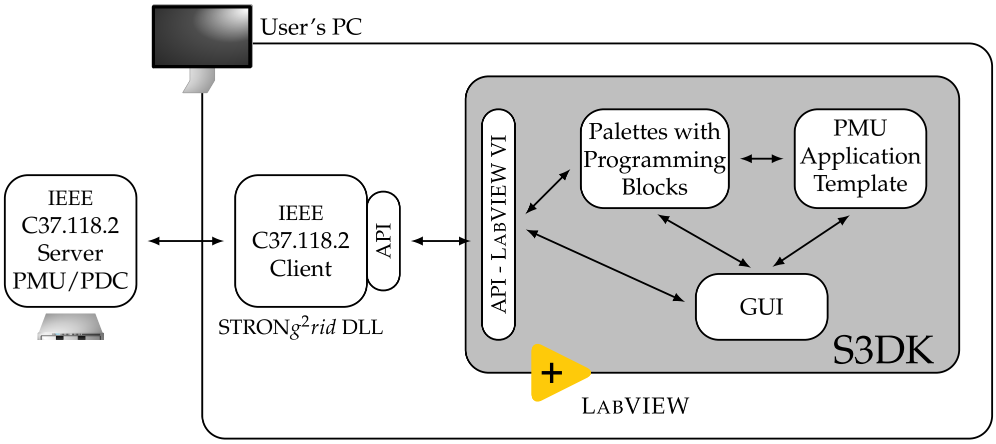

Figure 3. S3DK Main User Interface

comprehensive background in computer science and programming, to do any sort of rapid prototyping of synchrophasor applications. The academic community is more familiar with high level programming languages that abstract much of the programming complexity, such as MatLab, and LabVIEW. In a attempt to make the application development the most accessible to the power system research community, it was decided to develop a software development kit (SDK) dedicated to synchrophasor applications. This vision was shared by Statnett SF, the Norwegian Transmission System Operator, who supported the development through different research projects [29].

The Smart-grid Synchrophasor SDK (S3DK) was developed for the LabVIEW programming language that provides an environment facilitating the development of computationally advanced applications with highly interactive graphical user interfaces (GUIs). While both MatLab, and LabVIEW offer an environment with powerful toolboxes for advanced signal processing and computation functionalities, we selected the latter for its easy approach to designing GUIs. This is of particular importance for the target audience that is the power system academic community for fast prototyping of new idea and concepts with minimal training. Furthermore, LabVIEW recently released a community edition of its development suite that is free to use for non commercial use, expanding the target audience of the SDK to additional potential users in the power engineering community[2]. Applications can also be compiled as standalone programs to be shared with external partners for an easy deployment (see: https://www.ni.com/en/support/documentation/supplemental/19/introduction-to-the-labview-application-builder.html).

S3DK has three main components that are presented in the following Sections. The main goal is to present in summary fashion the most important components, while a comprehensive description of all the components and methods included in S3DK is available online at: https://alsetlab.github.io/S3DK/

3.1. Low-Level Communication

As described in the Section 2.2.2, the decentralized paradigm relies on a real-time data mediator (RTDM) to receive a stream of measurements, and to decode the communication protocol to deliver the data into a development environment. To this end, the S3DK leverages the STRONg^2*rid* Library [22], a RTDM that handles low-level communication tasks and decodes IEEE C37.118.2 protocol, in addition to provide additional low-level functionalities. This library is comprised by a separate $C++$ module that is compiled into a Dynamic Link Library (DLL) and used by the other components of the SDK.

[2] Since April 2020, National Instruments has released the LabVIEW Community Edition which is free of cost (for non-commercial use), which makes the use of S3DK a cost-free alternative for its potential users that lack a license for LabVIEW under certain conditions. For details see: https://www.ni.com/en/support/documentation/supplemental/20/labview-community-edition-usage-details.html.

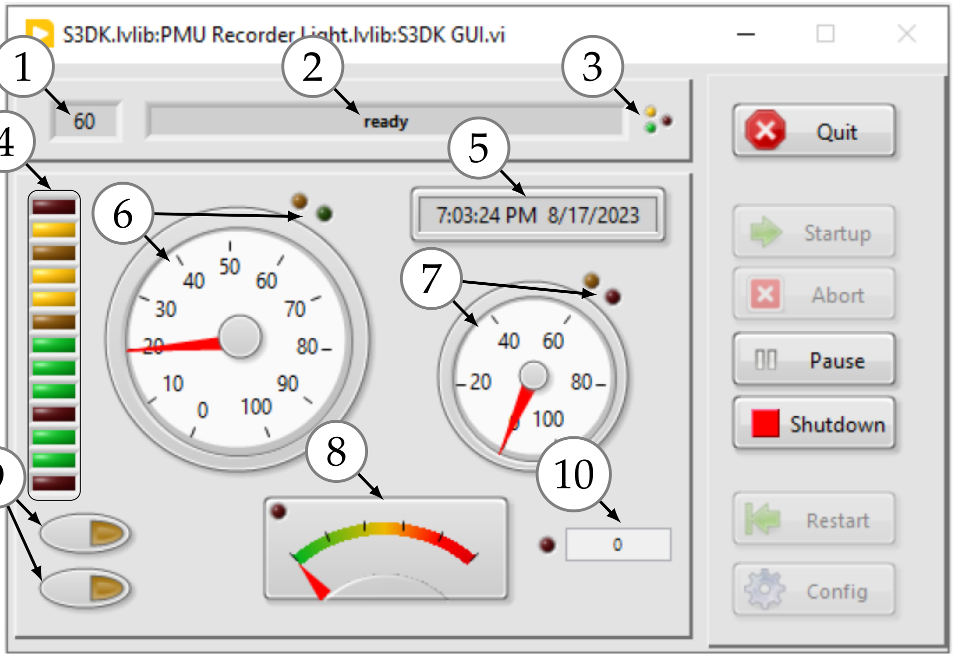

Figure 4. S3DK Main User Interface

The STRONg^2*rid* library implements all the functionalities of an IEEE C37.118.2 client to connect to a PMU / PDC stream(s) and to manage that connection. Furthermore, it was developed to be extended to manage parallel connections to multiple measurement streams synchronously, and streaming functionalities to emulate a PDC. The library handles the communication and continuous polling of the measurement data, and an Application Programming Interface (API) is implemented to make the measurements accessible into a development environment. The API features a set of LabVIEW specific methods as it was developed in conjunction with the rest of S3DK, but also features non specific methods and can easily be extended to be used in other environments.

3.2. Graphical UI

The main user-facing component of S3DK is a LabVIEW *Virtual instrument* (VI) that implements the measurements data handling establishing a connection to a PDC stream and creating data buffers for further delivery in the development environment (i.e., LabVIEW). It has a GUI, depicted on Fig. 4, that lets the user setup the PDC stream, and configure some basic settings for the data handling such as batch size and various timeouts for late data and connection loss. The remainder of the interface is dedicated to managing the connection to the data stream, including some visualization of the activity level:

1. Data Rate in samples per second,
2. Software state machine status,
3. General activity indicators,
4. Connection status indicators,
5. Timestamp of latest data frame received,
6. Data Frame processing activity indicators,
7. Access buffer writing activity indicators,
8. Bad data rate indicator,
9. Buttons to open additional windows for channel selection and seeing raw data,
10. Bad data samples counter.

In the background, the VI orchestrates the communication with the STRONg^2*rid* library through its API to connect to the PDC stream collect the data in real-time. It organizes the data received into a user configurable access buffer that can store multiple hours of data, and a queue that allows real-time access to the latest unread data.

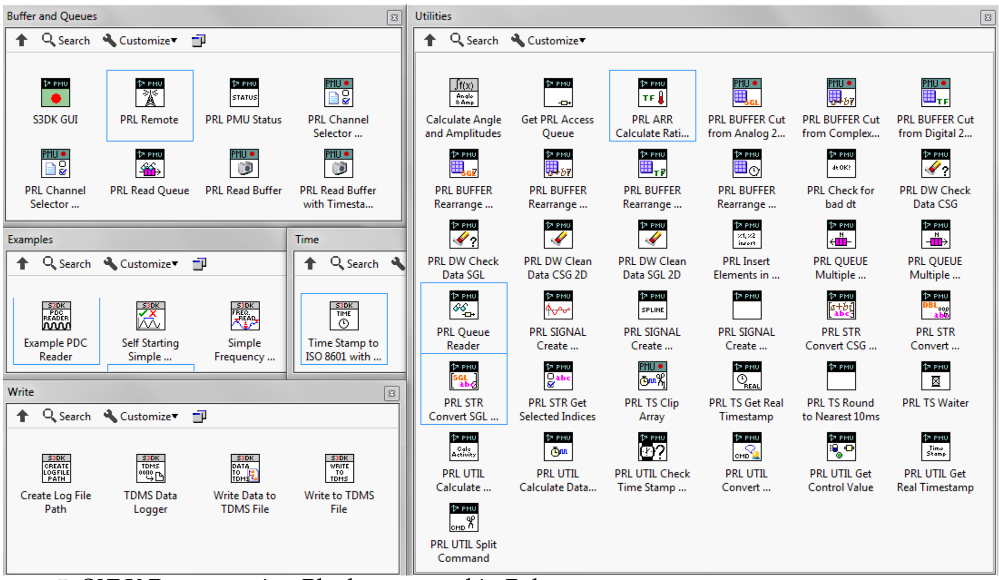

Figure 5. S3DK Programming Blocks arranged in Palettes

The VI can be used manually through its GUI, or can be integrated in an automated workflow through a remote facility that is accessible through a separate VI that lets the user trigger all the available user inputs pragmatically.

3.3. Palettes and Programming Functionalities

S3DK was developed to provide a real-time data mediator for developing PMU applications. It was developed to be integrated into the LabVIEW development environment as "palettes" (i.e., collections of VIs, akin to a MatLab toolbox or Python Library) organized into different sub-palettes by topic, shown in Fig. 5. The core functionalities of S3DK are embodied by the Graphical UI presented in the previous Section, and a few other VIs that allow its remote control, and the access to the access buffer and the queue for measurement delivery. These are grouped in the "Buffer and Queues" sub-palette.

A set of examples that demonstrate how to use the core functionalities from the "Buffer and Queues" sub-palette is available in the "Examples" sub-palette.

The development of these core functionalities required the development of several helper functions that were packaged into separate VIs. These include low level VIs that integrate with the STRONG^2*rid* library API, as well as a set of data manipulation VIs. The latter were deemed useful for other applications and integrated in the "Utilities" sub-palette.

The access buffer of the S3DK offers access to the measurements over a certain time horizon that is defined in the S3DK settings. The functionality relies on the LabVIEW TDMS file format to store the received measurements. The data logging was implemented in VIs, making the specific data structure used by S3DK available for the user in the "Write" sub-palette.

The "Time" sub-palette has a single VI converting the LabVIEW timestamp datatype into an ISO 8601 compliant string of characters.

Finally the last sub-palette is dedicated to a few examples that demonstrate the use of the VIs to access the measurements and remote control GUI.

4. Templates and Research Applications

The development of S3DK was quickly followed by the development of multiple PMU applications within the second author's research group. The access to real-time PMU measurements allowed to test new concepts and ideas for grid monitoring and control applications. We present in this Section a few notable examples.

4.1. Templates

Templates were the first applications that were developed. A few examples are directly included with S3DK to demonstrate the use of the main blocks for simple applications and automation.

In order to further lower the entry barrier to developing and testing new ideas, we decided to develop a "Template PMU App" that provides the basis for developing generic PMU applications. This allows researchers to focus on the core algorithm to develop, rather than the data ingestion. The template is built upon LabVIEW's own "Continuous Measurement and Logging" sample project that provides a framework for acquiring data, logging it and processing it. Thus, the template was prepared by integrating S3DK in the data acquisition loop, using its queue functionality get the latest unread data.

In the original template, the logging and processing loops are independent tasks with the measurements being sent independently to each loop. In the PMU application template, the logging task was replaced by a dedicated buffer that can store a chosen number of PMU measurements. The motivation of creating an additional buffer bypassing the built-in buffer of S3DK arose from the speed of execution of the the built-in version that uses storage into a local file to store the buffered measurements. This method allows to store large amounts of data, but its access time was not suited to sub-second access due to the need to read the data from a file stored on disk. The buffer provided in the template relies on storing the measurement data in the memory, keeping its access time very short. Furthermore a index functionality was integrated to allow tracking of which data has already been read by the processing loop(s), as well as some highly customizable data delivery method (e.g., unread data, past X samples, etc.)

Another contribution of this template app was to integrate a file reader to replay measurements saved into a CSV file. It was integrated in the data acquisition loop where the user can switch between reading a file or acquiring real-time data from S3DK. The template automates the flushing of the data currently in the buffer to replace with the new selected data, and automates the data reading to simulate "real-time" replay of the recorded measurement data akin to a live stream.

The template PMU app is distributed as a LabVIEW project as shown on Fig. 6

4.2. Monitoring Applications

As the first users of the S3DK, the research team led by the second author (graduate students, research engineers, post-docs, etc.) implemented a series of prototype applications demonstrating the different usages of PMU measurements for monitoring the grid.

Inspired by traditional monitoring solutions from grid control centers traditionally using SCADA measurements, a dashboard including a map of the Nordic Countries was developed to display real-time measurements of the system coming from PMUs that were deployed as part of the STRONg^2*rid* project[29]. The dashboard was primarily designed to display the frequency measured at different location, with configurable alarms for values exceeding the normal range of operation, i.e., [49.95 - 50.05 Hz]. A remote panel was also implemented by taking advantage of LabVIEW's built-in facilities for streaming data through a standard internet browser, as well as an iPad data dashboard.

Further developments focused on more advanced and more computationally intensive applications with the following applications:

- Monitoring Tool: a tool developed to detect power oscillations (e.g inter-area, sub-synchronous, etc.). In particular, it was applied to a case of detection of wind-farm oscillations in [7].
- Mode Estimation: a tool developed to harness ambient measurements for small-signal stability monitoring by estimating the grid's modes of oscillation [30].
- Kalman Filter: a tool implementing a method for extracting steady state information from synchrophasor measurement, as presented in [31].

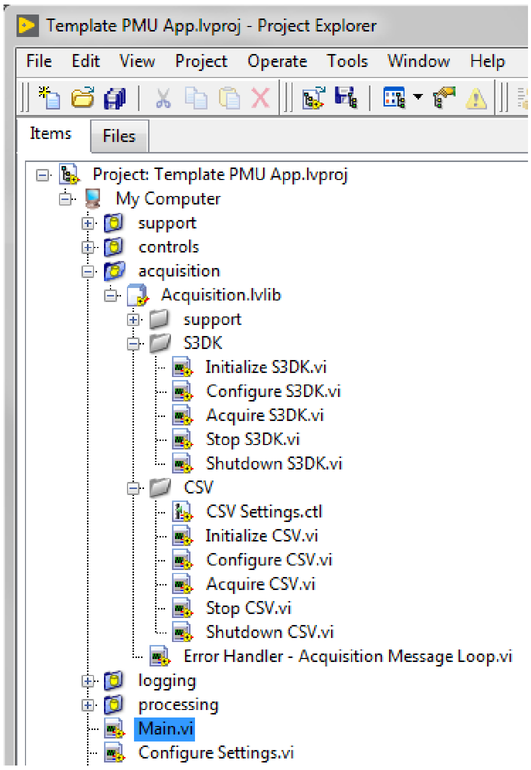

Figure 6. Template PMU Application Project File Explorer

- Steady-State Model Synthesis: a tool that uses the Kalman Filter and estimates a T-equivalent circuit between a number of PMUs, being validated in a real-world distribution network in Switzerland [32].
- Voltage stability monitor: is a tool that allows measuring the voltage stability of distribution networks, being able to recognize if the instability is due to the transmission or distribution network [33].
- Feeder Dynamic Rating: is an application that uses PMU data in combination with other sensor data to provide dynamic ratings of distribution feeders [34].

4.3. Control and Protection Applications

The early applications development focused on monitoring applications as these only require a single type of communication: a stream of measurements coming from the PDC that is handled by S3DK. As the research team developed the aforementioned applications and subsequently got more acquainted with S3DK, and the LabVIEW development environment, control and protection applications were the logical next step.

Note that control and protection applications not only need to read measurements, but also need a way to send control signals out to the power system (or its digital twin in the case of a simulation environment). This secondary communication channel is not part of S3DK, however, the S3DK served to acquire and process the data before being sent into different control loops, see [35] for details.

Notable examples of control applications are those dedicated to supplementary damping controls, including damping via load modulation and synchronous generator excitation system-based [35], and a STATCOM-based wide-area stabilizers in [36]. In addition, it has been used in testing a wide-area control system used as a transient stability booster [35].

Meanwhile, sample protection applications include an islanding detection and isolation scheme, and a synchronization scheme presented in [37], an automated microgrid clustering scheme in [38], an auto-reclosing scheme for active distribution networks [39], and a PMU-data assisted over-current protection device for feeders with solid-state transformers [40].

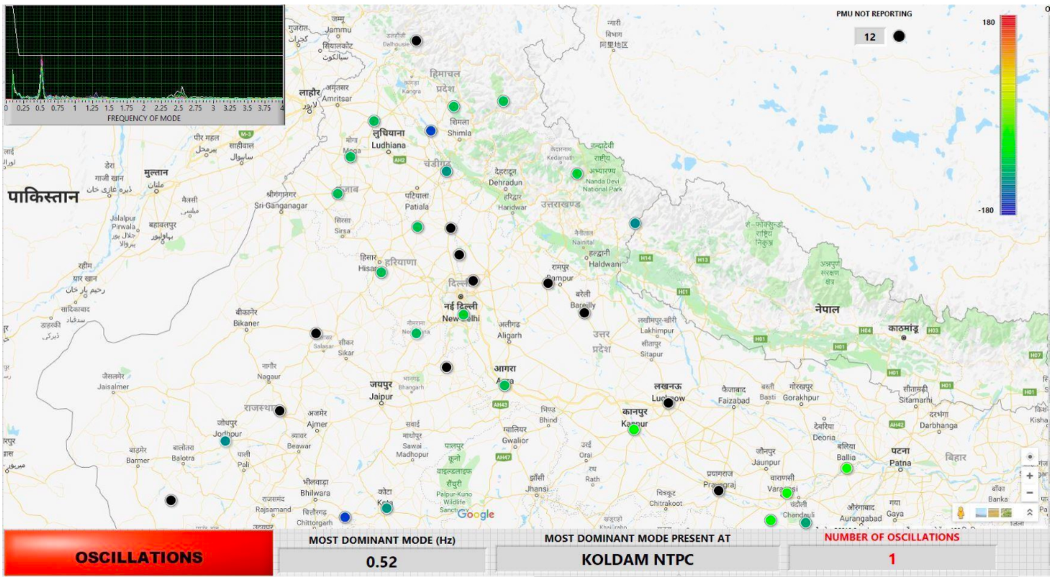

Figure 7. Oscillation Monitoring Dashboard

5. Industry Applications

5.1. Applications Developed at by Grid Controller of India Ltd.

Engineers from the Grid Controller of India Ltd's Northern Regional Load Dispatch Centre (NRLDC) leveraged S3DK to develop a set of monitoring applications, each answering a specific problem that presented itself in the operation of the Indian power grid.

5.1.1. Oscillation Monitoring

The main development focused on an application to detect inter-area oscillations, as depicted on Fig. 7. The application monitors oscillations happening in the frequency range [0.1 - 4] Hz, which covers inter-area modes and local modes. The main dashboard features an indicator that turns from green to red when oscillations are detected, with two accompanying displays showing the frequency of the dominant mode and the substation where the amplitude is detected to be the highest. The remainder of the GUI is dedicated to a map of the area being monitored by 35 PMUs in the north of India, with a dot indicating the position of each PMU. The dots are color coded to represent the angle information from the detected mode of oscillation, giving visual cues as to which buses in the system are coherent during a low frequency oscillation event.

Thanks to S3DK, three different methods for oscillations detection were evaluated: Fast Fourier Transform (FFT), Prony Analysis, and Estimation of Signal Parameters by Rotational Invariance Technique (ESPRIT) [41]. Upon testing it was found that the Prony method was highly sensitive to noise, leading to many false positives, and while the other methods performed similarly, the FFT method was the most computationally efficient and retained as main implementation. The frequency spectrum computed for each PMU is displayed on a detailed view of the application as depicted on Fig. 8. More details on this application can be found in [42].

5.1.2. Other Applications

Additional applications were developed by the same engineering team in an attempt to assist the configuration and calibration of various protection schemes in the grid. Differently from the previous section, these applications were develop as prototypes that are not yet in the production environment:

- Overflux Monitoring: The application displays the V/F value (as ratio of actual/nominal) of selected buses in real-time, as shown on Fig. 9. An early warning alerts the operator when this ratio passes 1.10 (the overflux protection trips at 1.14).

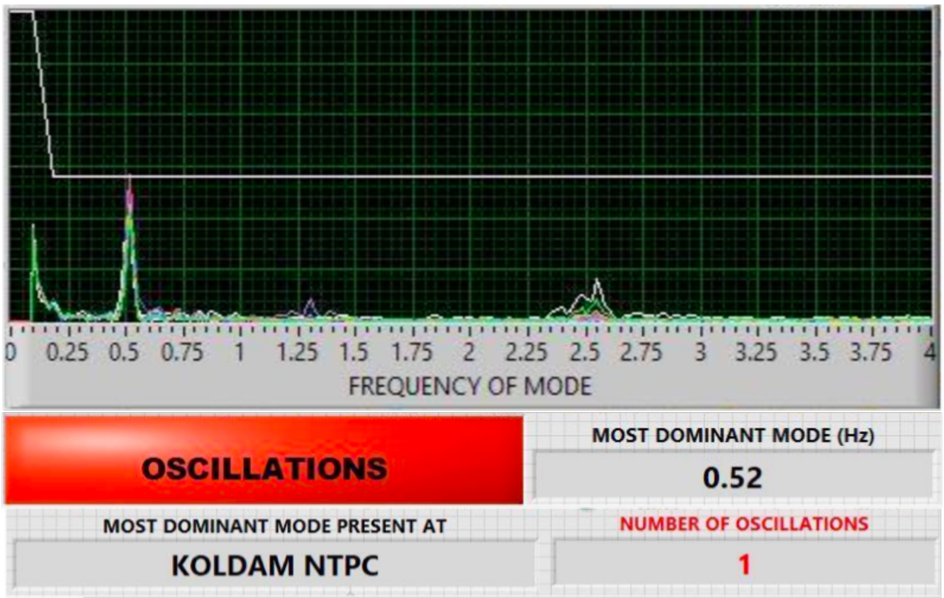

Figure 8. Oscillation Monitoring Details Window

- Load Point Tracking: This application can display the load point (measured impedance) of a selected transmission line end in real-time in a polar diagram as shown on Fig. 10. This allows to monitor that the load point doesn't encroach the protection zone.
- Zero sequence voltage: This application displays the zero sequence voltage of a selected bus on a polar plane as shown on Fig. 11. It is helpful to monitor unbalances at the selected point of monitoring.

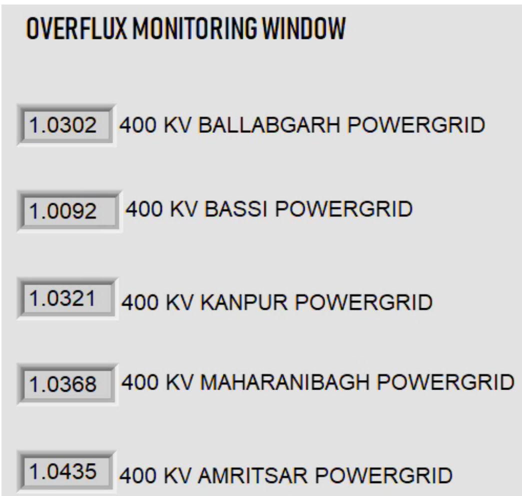

Figure 9. Transmission Line Overflux Monitoring Application

5.2. Other Industrial Applications

As described in [43], Statnett SF has used an early prototype of the S3DK for the development of several voltage stability and oscillation monitoring applications, while at the same time being particularly useful for the monitoring of High-Voltage direct current (HVDC) lines. Even though these application prototypes have not been deployed for regular use of operators, the authors of [43] highlight that they have helped in shaping the tools that are being used in the control room.

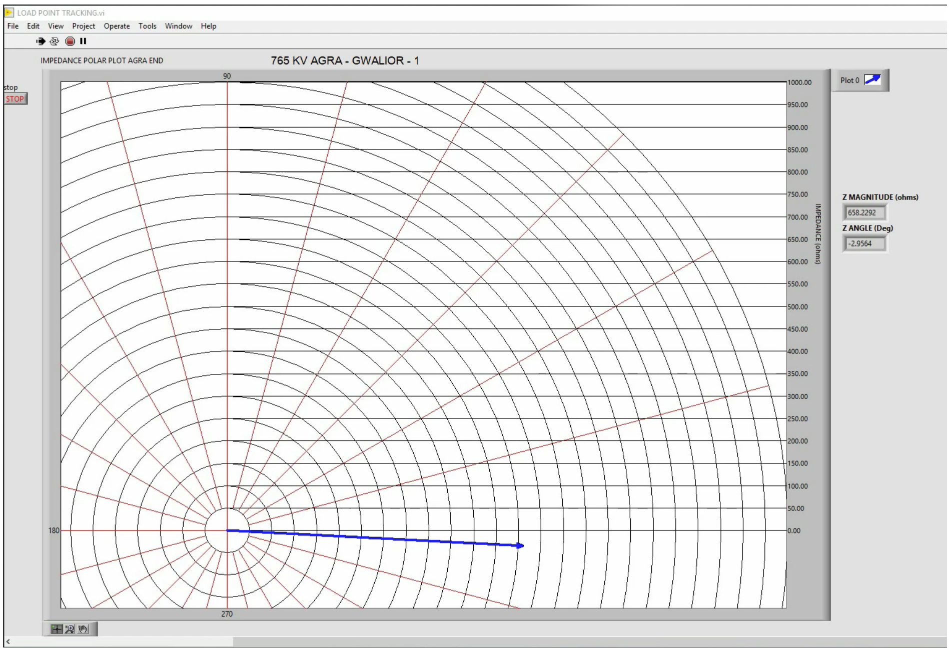

Figure 10. Load Point Tracking Application

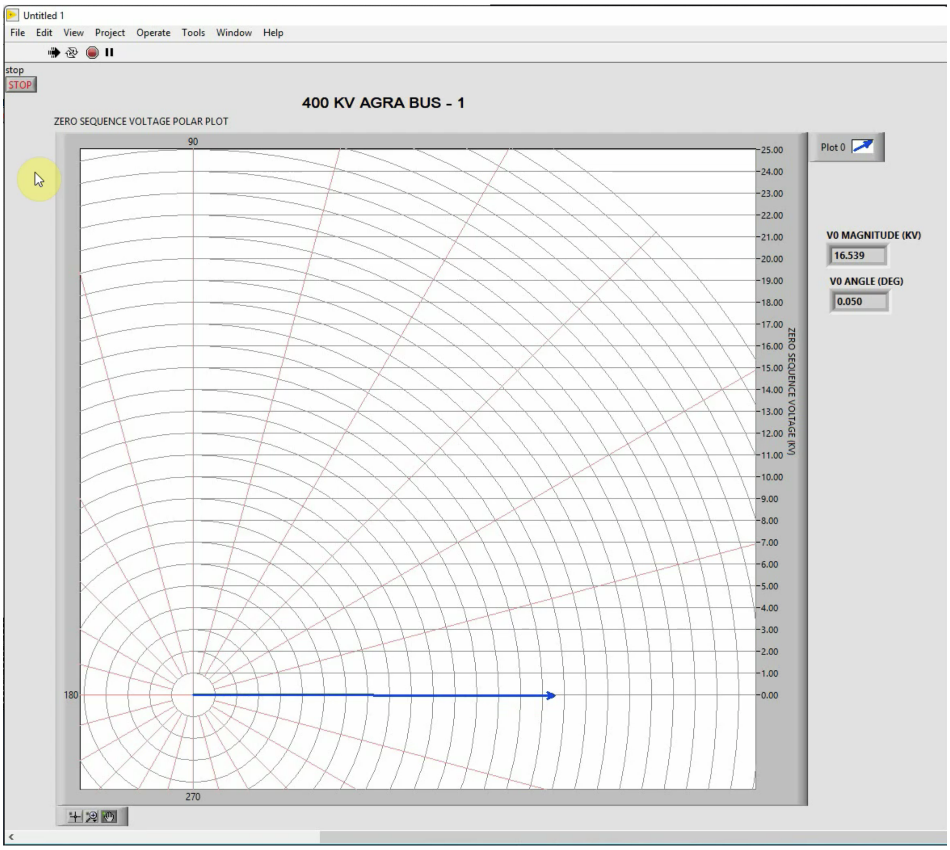

Figure 11. Zero Sequence Monitoring Application

6. Conclusion

S3DK is a toolkit The S3DK is a toolkit that was developed to facilitate the fast prototyping of synchrophasor applications. This paper aims to promote its release as an open source project and to summarize its virtues and uses, so that other researchers may exploit it for their own purposes. This paper also shows that the toolkit was instrumental in supporting the research activities having been used to develop a wide array of new applications covering monitoring, control and protection application, which are documented in separate publications cited in this paper. Lastly, an engineering team from an the Grid Controller of India Ltd., an Indian grid operator, selected S3DK to develop applications answering a set of problems faced in the control center to facilitate the operators' work of operating the grid with minimal disturbances. It is important to note that these applications were developed independently from the developers of the S3DK, which shows the potential of S3DK in facilitating PMU prototyping in industry.

Acknowledgments: The authors would like to thank the following individuals of Statnett SF, the Norwegian Transmission System Operator: Stig Løvlund, L. K. Vormedal, and the late Jan Ove Gjerde who supported the development of this software towards its open source release through different research projects. The authors would also like to recognize the contributions of Vemund H. Aarstrand of Enfo (LabVIEW development), Stig Garberg of CodeSmiths AS (C++ development), Jordan Grey and Emmett Williamson (online documentation development).

Data Availability Statement: he code of the S3DK, and its precompiled LabVIEW packages are openly available in Github at https://github.com/ALSETLab/S3DK.

Conflicts of Interest: Author Shashank Tyagi was employed by the company EirGrid Group. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Mittelstadt, W.A.; Krause, P.E.; Wilson, R.E.; Overholt, P.N.; Sobajic, D.J.; Hauer, J.F.; Rizy, D.T. The DOE Wide Area Measurement System (WAMS) Project: Demonstration of dynamic information technology for the future power system. Technical Report CONF-960434-1, USDOE Bonneville Power Administration, Portland, OR (United States), 1996.
2. Hauer, J.; Bhatt, N.; Shah, K.; Kolluri, S. Performance of "WAMS East" in providing dynamic information for the North East blackout of August 14, 2003. In Proceedings of the IEEE Power Engineering Society General Meeting, 2004., 2004, pp. 1685–1690 Vol.2.
3. Cai, J.; Huang, Z.; Hauer, J.; Martin, K. Current Status and Experience of WAMS Implementation in North America. In Proceedings of the 2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific, 2005, pp. 1–7. ISSN: 2160-8644.
4. Hauer, J.F.; Trudnowski, D.J.; DeSteese, J.G. A Perspective on WAMS Analysis Tools for Tracking of Oscillatory Dynamics. In Proceedings of the 2007 IEEE Power Engineering Society General Meeting, 2007, pp. 1–10. ISSN: 1932-5517.
5. Usman, M.U.; Faruque, M.O. Applications of synchrophasor technologies in power systems. *Journal of Modern Power Systems and Clean Energy* **2019**, *7*, 211–226. https://doi.org/10.1007/s40565-018-0455-8.
6. Abo-Khalil, A.G. Impacts of Wind Farms on Power System Stability. In *Modeling and Control Aspects of Wind Power Systems*; Muyeen, S.M.; Al-Durra, A.; Hasanien, H.M., Eds.; IntechOpen: Rijeka, 2013; chapter 7. https://doi.org/10.5772/55090.
7. Vanfretti, L.; Baudette, M.; White, A.D. Chapter 31 - Monitoring and Control of Renewable Energy Sources using Synchronized Phasor Measurements. In *Renewable Energy Integration (Second Edition)*; Jones, L.E., Ed.; Academic Press: Boston, 2017; pp. 419–434.
8. Saleem, M.; Saha, S.; Roy, T.K.; Ghosh, S.K. Assessment and management of frequency stability in low inertia renewable energy rich power grids. *IET Generation, Transmission & Distribution* **2024**, pp. n/a–n/a. https://doi.org/10.1049/gtd2.13129.
9. Pierrou, G.; Lai, H.; Hug, G.; Wang, X. A Decentralized Wide-Area Voltage Control Scheme for Coordinated Secondary Voltage Regulation Using PMUs. *IEEE Transactions on Power Systems* **2024**, *PP*, 1–13. https://doi.org/10.1109/TPWRS.2024.3381553.
10. Priyadarshi, A.; Yadav, B. Bad data detection and multi-level classification of events in synchrophasor measurements using AXGBoost algorithm. *Electrical Engineering* **2023**, *106*, 1–15. https://doi.org/10.1007/s00202-023-01963-8.
11. MansourLakouraj, M.; Gautam, M.; Livani, H.; Benidris, M. A multi-rate sampling PMU-based event classification in active distribution grids with spectral graph neural network. *Electric Power Systems Research* **2022**, *211*, 108145. https://doi.org/10.1016/j.epsr.2022.108145.
12. Wang, C.; Qin, Z.; Hou, Y.; Yan, J. Multi-Area Dynamic State Estimation With PMU Measurements by an Equality Constrained Extended Kalman Filter. *IEEE Transactions on Smart Grid* **2016**, *PP*, 1–1. https://doi.org/10.1109/TSG.2016.2570943.
13. Schweitzer Engineering Laboratories Inc.. Synchrowave Operations. Available online at: https://selinc.com/solutions/software/synchrowave-operations/, last accessed on 04.09.2024.
14. Electric Power Group. Real-Time Dynamics Monitoring System (RTDMS). Available online at: https://www.electricpowergroup.net/epg_products/rtdms/, last accessed on 04.09.2024.
15. General Electric. Phasor Point. Available online at: https://www.ge.com/digital/applications/transmission/wide-area-management-system-and-control-wams-wamc, last accessed on 04.09.2024.

16. Ma, J.; Venkatasubramanian, M.V.; Feuerborn, S.; Black, C.; Halpin, M.; Hsu, S.M. A Software Suite for Power System Stability Monitoring Based on Synchrophasor Measurements. In *Power System Grid Operation Using Synchrophasor Technology*; Nuthalapati, S.N., Ed.; Power Electronics and Power Systems, Springer International Publishing: Cham, 2019; pp. 449–476.
17. Office of Electricity, Department of Energy. Project Grant DEOE0000700. Available online at: https://www.highergov.com/grant/DEOE0000700/, last accessed on 04.09.2024.
18. Rosso, A. Demonstration of a Novel Synchrophasor-based Situational Awareness System: Wide Area Power System Visualization, On-line Event Replay and Early Warning of Grid Problems. Technical report, Electric Power Research Institute, Incorporated, 2012.
19. Robertson, R. Open and Extensible Control & Analytics Platform for Synchrophasor Data. Technical Report DOE-GPA-0778-1, Grid Protection Alliance, Inc., 2018.
20. Madani, V.; Giri, J.; Kosterev, D.; Novosel, D.; Brancaccio, D. Challenging Changing Landscapes: Implementing Synchrophasor Technology in Grid Operations in the WECC Region. *IEEE Power and Energy Magazine* **2015**, *13*, 18–28. https://doi.org/10.1109/mpe.2015.2431213.
21. IEEE Standard for Synchrophasor Data Transfer for Power Systems, 2011. Pages: 1-53 Publication Title: IEEE Std C37.118.2-2011 (Revision IEEE Std C37.118-2005), https://doi.org/10.1109/ieeestd.2011.6111222.
22. Baudette, M.; Firouzi, S.R.; Vanfretti, L. The STRONgrid library: A modular and extensible software library for IEEE C37.118.2 compliant synchrophasor data mediation. *SoftwareX* **2018**, *7*. https://doi.org/10.1016/j.softx.2018.08.001.
23. 2020 Smart Grid System Report. Technical report, United States Department of Energy, 2022.
24. Tu, H.; Du, Y.; Yu, H.; Dubey, A.; Lukic, S.; Karsai, G. Resilient Information Architecture Platform for the Smart Grid: A Novel Open-Source Platform for Microgrid Control. *IEEE Transactions on Industrial Electronics* **2020**, *67*, 9393–9404. https://doi.org/10.1109/TIE.2019.2952803.
25. Zuo, J.; Carroll, R.; Trachian, P.; Dong, J.; Affare, S.; Rogers, B.; Beard, L.; Liu, Y. Development of TVA SuperPDC. In Proceedings of the 2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008, pp. 1–8. ISSN: 1932-5517.
26. Melton, R.B.; Schneider, K.P.; Lightner, E.; Mcdermott, T.E.; Sharma, P.; Zhang, Y.; Ding, F.; Vadari, S.; Podmore, R.; Dubey, A.; et al. Leveraging Standards to Create an Open Platform for the Development of Advanced Distribution Applications. *IEEE Access* **2018**, *6*, 37361–37370. https://doi.org/10.1109/ACCESS.2018.2851186.
27. Grid Protection Alliance. OpenPDC. Available online at: https://github.com/GridProtectionAlliance/openPDC, last accessed on 04.09.2024.
28. Naglic, M.; Popov, M.; Meijden, M.A.M.M.v.d.; Terzija, V. Synchro-Measurement Application Development Framework: An IEEE Standard C37.118.2-2011 Supported MATLAB Library. *IEEE Transactions on Instrumentation and Measurement* **2018**, *67*, 1804–1814. https://doi.org/10.1109/TIM.2018.2807000.
29. Almas, M.; Baudette, M.; Vanfretti, L.; Lovlund, S.; Gjerde, J. Synchrophasor network, laboratory and software applications developed in the STRONg^{2}rid project. In Proceedings of the IEEE Power and Energy Society General Meeting, 2014, Vol. 2014-Octob. https://doi.org/10.1109/PESGM.2014.6938835.
30. Peric, V.S.; Baudette, M.; Vanfretti, L.; Gjerde, J.O.; Lovlund, S. Implementation and testing of a real-time mode estimation algorithm using ambient PMU data. In Proceedings of the 2014 Clemson University Power Systems Conference. IEEE, 2014, pp. 1–5. https://doi.org/10.1109/PSC.2014.6808116.
31. Mahmood, F.; Hooshyar, H.; Vanfretti, L. A method for extracting steady state components from Syncrophasor data using Kalman Filters. In Proceedings of the 2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC). IEEE, 2015. https://doi.org/10.1109/eeeic.2015.7165393.
32. Mahmood, F.; Vanfretti, L.; Pignati, M.; Hooshyar, H.; Sossan, F.; Paolone, M. Experimental Validation of a Steady State Model Synthesis Method for a Three-Phase Unbalanced Active Distribution Network Feeder. *IEEE Access* **2018**, *6*, 4042–4053. https://doi.org/10.1109/ACCESS.2018.2792320.
33. Bidadfar, A.; Hooshyar, H.; Monadi, M.; Vanfretti, L. Decoupled voltage stability assessment of distribution networks using synchrophasors. In Proceedings of the 2016 IEEE Power and Energy Society General Meeting (PESGM), Boston, MA, USA, 2016; pp. 1–5. https://doi.org/10.1109/PESGM.2016.7741644.
34. Singh, N.; Hooshyar, H.; Vanfretti, L. Feeder dynamic rating application for active distribution network using synchrophasors. *Sustainable Energy, Grids and Networks* **2017**, *10*, 35–45. https://doi.org/10.1016/j.segan.2017.02.004.
35. Vanfretti, L.; Jónsdóttir, G.; Almas, M.; Rebello, E.; Firouzi, S.; Baudette, M. Audur—A platform for synchrophasor-based power system wide-area control system implementation. *SoftwareX* **2018**, *7*, 294–301. https://doi.org/10.1016/j.softx.2018.08.003.
36. Rebello, E.; Vanfretti, L.; Almas, M.S. Experimental Testing of a Real-Time Implementation of a PMU-Based Wide-Area Damping Control System. *IEEE Access* **2020**, *8*, 25800–25810. https://doi.org/10.1109/ACCESS.2020.2970988.
37. Almas, M.; Vanfretti, L. A Hybrid Synchrophasor and GOOSE-Based Power System Synchronization Scheme. *IEEE Access* **2016**, *4*, 4659–4668. https://doi.org/10.1109/ACCESS.2016.2601445.
38. Monadi, M.; Hooshyar, H.; Vanfretti, L.; Mahmood, F.; Candela, J.I.; Rodriguez, P. Measurement-Based Network Clustering for Active Distribution Systems. *IEEE Transactions on Smart Grid* **2019**, *10*, 6714–6723. https://doi.org/10.1109/TSG.2019.2910510.

39. Monadi, M.; Hooshyar, H.; Vanfretti, L. Design and real-time implementation of a PMU-based adaptive auto-reclosing scheme for distribution networks. *International Journal of Electrical Power & Energy Systems* **2019**, *105*, 37–45. https://doi.org/10.1016/j.ijepes.2018.07.064.
40. Hooshyar, H.; Baran, M.; Firouzi, S.; Vanfretti, L. PMU-assisted overcurrent protection for distribution feeders employing Solid State Transformers. *Sustainable Energy, Grids and Networks* **2017**, *10*, 26–34. https://doi.org/10.1016/j.segan.2017.02.003.
41. Ray, P. Power system low frequency oscillation mode estimation using wide area measurement systems. *Engineering Science and Technology, an International Journal* **2017**, *20*, 598–615. https://doi.org/10.1016/j.jestch.2016.11.019.
42. Shashank, T.; Manoj, A.; Alok, K.; Banerjee, S.; Nallarasan, N. A tool to detect Low frequency power system oscillations in real time using PMU data. In Proceedings of the CIGRE Session 2022 . CIGRE (International Council on Large Electric Systems) , Aug 2022.
43. Karlsen, D.; Uhlen, K.; Vormedal, L. Introducing PMU-based Applicatons in the Control Room Setting. In Proceedings of the CIGRE Session 2018 . CIGRE (International Council on Large Electric Systems) , Aug 2018.