On Investigating the Benefits of TTCN-3-Based Testing in the Context of IEC 61850

Georg Panholzer, Christof Brandauer
Advanced Networking Center
Salzburg Research Forschungsgesellschaft mbH
Salzburg, Austria
email: {firstname.lastname}@salzburgresearch.at

Stephan Pietsch
Testing Technologies IST GmbH
Berlin, Germany
email: pietsch@testingtech.com

Jürgen Resch
Ing. Punzenberger COPA-DATA GmbH
Salzburg, Austria
email: JuergenR@copadata.com

***Abstract*—This paper is concerned with approaches to testing in the context of the International Electrotechnical Commission (IEC) standard IEC 61850, which is gaining momentum in general power utility automation tasks. We outline the current state-of-the-art in IEC 61850 testing and argue that an approach based on the Testing and Test Control Notation (TTCN-3), as has been used successfully in other industries, would provide several advantages. The test specification and execution language TTCN-3 is briefly introduced and potential benefits are discussed. We describe how we used TTCN-3 to implement conformance tests for an IEC 61850 client and run those tests against a certified server. Finally, we present a performance test case for synchrophasor measurements on the basis of distributed TTCN-3.**

Keywords*–*IEC 61850; TTCN-3; conformance testing; performance testing.

I. Introduction

The world's energy systems are currently undergoing radical changes that affect individuals, as well as the society as a whole. The change is provoked by a multitude of factors like the limited availability of conventional energy sources, increased demand, increased energy costs, new legal regulations, and the renunciation of energy monopolies.

At the European level the '20-20-20' goals are strived to be reached until 2020: saving 20% of the EU's primary energy consumption, a binding target of 20% reduction of greenhouse gas emissions and 20% renewable energies. The increased integration of renewable energies poses new challenges. Until recently, electricity networks have been "one-way" streets, due to the unidirectional flow of energy from production via transmission and distribution to customers. Through the decentralized integration of renewable energy sources, volatile in their nature, bidirectional energy flows become a reality and the task of controlling these flows becomes much more challenging.

To cope with these challenges, the concept of a next generation electric power system, a so called "smart grid", has emerged. It is characterized by the heavy use of information and communication technologies (ICT), e.g., digital processing and communications, so that data flows and information management become an integral part of the future grid. Indeed, the "intelligence" of the smart grid as implemented in sophisticated energy applications is based on the (near) real-time exchange of measurement and control data amongst a large number of devices located throughout the whole grid [1].

The inter-dependent processes of data acquisition, communication and (often automated) control of the smart grid are highly complex and stringent requirements in terms of correctness, standard conformance, interoperability, performance, and security are posed on the components involved. The topic of testing plays a major role in meeting these requirements.

Our studies on the current state of testing are focused on the multi-part standard IEC 61850. Its development started in 1994 with the goal of developing a comprehensive world-wide standard for the design and operation of substation automation. The main requirements were to advance beyond a growing set of proprietary, incompatible and non-comprehensive approaches to communication solutions in substation automation and to define a global standard that facilitates interoperability and integration. While the initial focus was on the electric substation, the standard has been extended in several directions (substation to substation, substation to control center, hydroelectric power plants, distributed energy resources, wind turbines, etc.) and is nowadays employed more broadly for general power utility automation tasks.

This paper is divided in the following sections: Section II provides a short overview over the multi-part standard IEC 61850, Section III describes the International Organization for Standardization (ISO)/International Telecommunication Union (ITU)/European Telecommunications Standards Institute (ETSI) approach to testing and briefly introduces TTCN-3. In Section IV, we describe how we used TTCN-3 for conformance testing of a IEC 61850 client and for performance testing of a IEC 61850-90-5 based synchrophasor transmission system. In the last section, we present our conclusion and plans for future work.

II. IEC 61850

IEC 61850 must not be seen as yet another communication protocol (like, e.g., many fieldbus protocols that have been invented over the course of time). In fact, IEC 61850 communications are based on existing and matured networking technologies (Ethernet, TCP/IP). A major strength lies in the information model and the device and vendor agnostic configuration and description language that are built upon it.

The information model is an object oriented model that contains a rich set of hierarchical classes which are used to describe physical devices, their functions, services, status information, measured values etc. in a standardized way (IEC 61850-7-3 / 61850-7-4). The information model enables the exchange of semantically well-defined information using common naming conventions. The way the information exchange takes place is abstractly defined in the form of an Abstract

 ISBN: 978-1-61208-406-0

TABLE I. TEST PROCEDURE 'CLIENT ASSOCIATION 1'.

cAss1	Associate and force a client to release a TPAA
Expected result: 1. SUT accepts Associate.response+ from server 2. SUT returns to "state" where it is able to start a new TPAA with the same server	
Test description: 1. Set-up a TPAA with one server 2. Force SUT to release or abort TPAA 3. Repeat step 1 and 2, 10 times server	

Communication Service Interface (ACSI, IEC 61850-7-2). The ACSI defines various types of communication services (e.g., directory services, read/write data (datasets), activate group settings, transmission of reports, file transfer, etc.) and furthermore specifies which of these services can be used with the specific elements of the information model. As an example, the `getServerDirectory` ACSI service can (naturally) only be used in combination with the `Server` object of the information model.

The definition of how the abstract services make use of communication protocols to actually transmit data on the network is defined in the so called Specific Communication Service Mappings (SCSM). By following this two-tiered approach the IEC 61850 communication services are decoupled from their concrete networking implementation. This affords a long-term stable interface towards applications without being locked into a specific networking technology. The first standardized mapping for client-server communication is based on the Manufacturing Message Specification (MMS) protocol [2][3], which makes use of OSI protocols on top of TCP/IP.

A. Testing in IEC 61850

Part 10 of the IEC 61850 standard *Conformance testing methodology and framework* is dedicated to conformance and performance testing. A conformance test has to include documentation and version control (IEC 61850-4), configuration (IEC 61850-6), data model (IEC 61850-7-3 and -7-4) and mapping of ACSI models and services (IEC 61850-7-2). Moreover, the testing "ecosystem" is explained including how to certify a tester. Part 10 identifies the areas to be tested (e.g., association, reporting, etc.) and introduces 167 server test cases for the ACSI mapping, all of which are mandatory if supported by the tested system. The positive and negative test cases are briefly described. Based on this framework, the testing subgroup of the Utility Communication Architecture International user group (UCAIug), a *"not-for-profit corporation of utility users and supplier companies"* [4] elaborated more detailed test procedures for servers as well as clients, performance of fast event distribution with Generic Object Oriented Substation Events (GOOSE) and extended server reports.

As a common denominator, all these test cases are described in prose using the table template as required by IEC 61850-10. As an example, Table I shows the description of a simple client association test case.

Such brief prose descriptions naturally leave a lot of room for interpretation and many aspects of the test, some of which have certainly been in mind by the test developers, are not present in the test cases. As an example, it is not specified how to verify that the association has indeed been established. Where are the Points of Observation and Control (POC)? What exactly are the criteria against which the expected Response+ reply has to be matched? Which protocol fields are required? Do some of them need to contain specific values? Which are irrelevant for the given test? It is obvious that the test cases, while still remaining on the abstract level, could be significantly enhanced by making use of a (more) formal description. We have thus studied other approaches to conformance testing with a special focus on communication protocols.

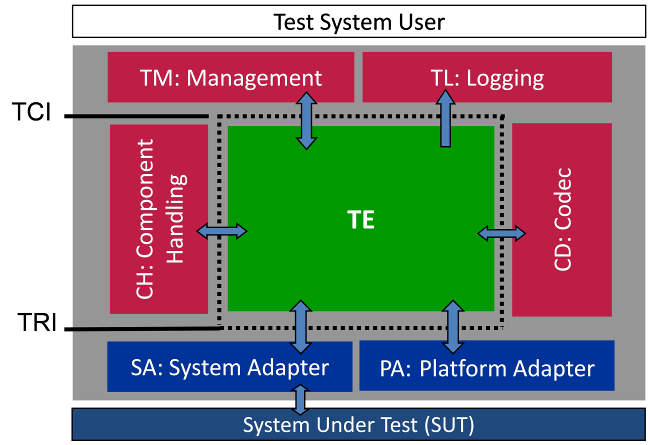

Figure 1. The TTCN-3 test system architecture.

III. THE ISO/ITU/ETSI APPROACH TO (CONFORMANCE) TESTING

Organizations like the ISO, ITU or ETSI are concerned with ensuring that different implementations of a specification conform to the recommendation or standard they are based upon. To this end they have developed the Conformance testing methodology and framework (CTMF) which is published as the ISO/IEC 9646 [5] and ITU-T X.290 to X.296 series of standards.

The CTMF is the standard method for testing OSI communication protocols and it is thus thematically close to IEC 61850 which also partly relies on OSI communication protocols.

A. Testing and Test Control Notation

Part 3 [6] of the ISO 9646 standard series introduces the tree and tabular notation (TTCN) for a formal description of abstract test cases. It is actively developed by ETSI and ITU-T. The current version is TTCN-3 [7] (now called Testing and Test Control Notation).

TTCN-3 has been successfully employed over a decade for major industrial testing efforts in the context of IPv6, SIP, WiMAX, LTE, TETRA, AUTOSAR, and others [8][9]. Initially, TTCN(-3) has been used exclusively to test communication protocols but it has since become a universal testing language that has established itself also in the automotive and medical domain.

TTCN-3 is a formal language which is specifically tailored to testing. As such it provides constructs that are not found in general purpose programming languages. An example are language elements for data matching that are commonly required for comparing incoming data against test expectations. The language provides an extensive type system and constructs for message-based and procedure-based communication. The concurrent execution of test components is a natural fit to the language. As the language is standardized and has a well-defined semantic for each and every element, a tool vendor lock-in situation is prevented. TTCN-3 is not “just” a language but instead provides a complete test system architecture with standardized interfaces among its components as depicted in Figure 1. There are TTCN-3 extensions related to, e.g., performance and real-time testing, as well as interfaces with continuous signals.

The abstract TTCN-3 test cases are compiled into executable code, which, in combination with a TTCN-3 runtime system, form the core Test Executable (TE). Codecs (CD) provide for the transformation between TTCN-3 data types and the native data format of the System Under Test (SUT). The Test Runtime Interface (TRI) [10] provides functions for communicating with the SUT via the System Adapter (SA) and the platform that hosts the test system via the Platform Adapter (PA). Implementation of the SA and PA components are needed to obtain an executable test system for a concrete SUT.

B. *Potential benefits of TTCN-3-based testing in IEC 61850*

From a technical point of view, we argue that there are several significant benefits in providing abstract test cases in TTCN-3 compared to prose descriptions. The standardized TTCN-3 language with its well-defined semantic helps eliminate the ambiguities of English (or any other language for that matter) and, consequently, a lot of room for interpretation. It allows for precise specifications of the expected static and dynamic behavior and provides a full-featured so called template system that enables brief and concise data matching statements. As TTCN-3 can make use of ASN.1 and XML Schema types, as well as IDL-based interface definitions, the type specification of standards can often be used directly in (or at least imported into) the TTCN-3 environment. In IEC 61850, the MMS standard – the currently used SCSM mapping for client-server communication – makes use of message types defined in ASN.1.

If abstract TTCN-3 tests were provided to the community, they would serve as a common, reusable starting point for any testing activities. Still, it would not restrict testers with respect to the choice of testing tools and/or implementation languages, respectively.

From a non-technical point of view, we argue that an open standard of a formal test description and execution language is preferable to prose test specifications and their derived proprietary and closed test implementations. Particularly for critical infrastructure like energy networks a review of test results is indispensable. In order to verify and reproduce those test results, it is necessary to make the test methodologies and surrounding conditions publicly available. Standardized TTCN-3-based test specifications readily fulfill a lot of these requirements and thus can significantly improve trust in the test results.

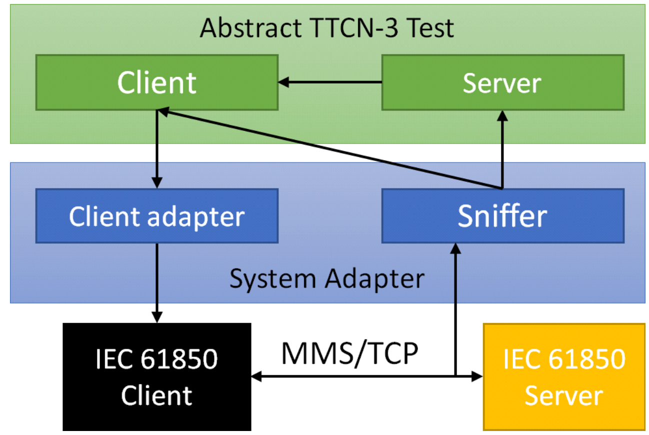

Figure 2. Setup for the IEC 61850 client conformance tests.

IV. IEC 61850 TESTING WITH TTCN-3

Interestingly, TTCN-3 has so far been seldom used in the context of energy networks [11][12]. As a consequence of the arguments listed above, the authors of this report are investigating the applicability of TTCN-3 for testing components and systems based on IEC 61850.

A. *IEC 61850 client conformance tests*

To start off, we implemented TTCN-3 test cases for a subset of the IEC 61850 client tests as defined by the UCAIug [13]. All tests were run against a certified server. The client has an API to trigger certain actions and query its status. We developed a system adapter that makes this API accessible via TTCN-3. On the server side, a network sniffer captures all network traffic and forwards it to the test runtime. The setup is depicted in Figure 2 and Listing 1 shows the relevant parts of an abstract test case for a client association test in TTCN-3.

Two *components*, `v_Server` and `v_Client` act as TTCN-3 proxies for the server and client respectively. They do not implement any real functionality but simply exchange messages within the runtime environment or with the actual implementations via a system adapter using so called *ports*. In our setup we have three ports, `pt_ObservationPort`, `pt_ControlPort` and `pt_CommPort`.

Both components receive messages from the sniffer via `pt_ObservationPort`. The sniffer’s system adapter uses the source IP address to forward the messages to the correct component. Messages with the source IP of the client are sent to the client component and messages with the server’s source IP are sent to the server component. The system adapter also maps from the on-wire data formats to the TTCN-3 data types.

The client component can access the client’s API via `pt_ControlPort`. The client’s system adapter takes the `requestPDU` (e.g., a *MMS initiate-RequestPDU* in case of the client association test case using the MMS SCSM) and calls the correct client API function.

The last port, `pt_CommPort`, connects the server component and the client component within the TTCN-3 environment. The server component uses that port to send the packets it received from the sniffer to the client.

Neither the server component nor the comm port are actually necessary, as the observation port could be used for

Copyright (c) IARIA, 2015. ISBN: 978-1-61208-406-0

```
testcase Ass1_Associate() runs on MTC system TSI {
    clientAddress := {ipAddress := "192.168.1.1"}
    serverAddress := {ipAddress := "192.168.1.2"}
    f_configClientServer(clientAddress, serverAddress);

    v_Server.start(f_Replay());
    v_Client.start(f_Associate(requestPDU, responsePDU));
    f_waitAndGuard();

    f_deconfigClientServer();
}

function f_Associate(in 61850Pdu requestPdu, template 61850
     Pdu responsePlusPdu) runs on ClientComponent  {
    pt_ObservationPort.send(Command:startTrace);
    pt_ControlPort.send(requestPdu);

    // wait for the association request packet
    alt {
        [] pt_ObservationPort.receive(requestPdu) {
            setverdict(pass);
        }
        [] pt_ObservationPort.receive(61850Pdu:?) {
            log("Ignoring Message");
            repeat;
        }
    }
    // wait for the association response packet
    alt {
        [] pt_CommPort.receive(responsePdu) {
            setverdict(pass);
        }
        [] pt_CommPort.receive(negativeResponsePdu) {
            setverdict(fail);
        }
        [] pt_CommPort.receive(61850Pdu:?) {
            log("Ignoring Message",);
            repeat;
        }
    }

    pt_ObservationPort.send(Command:stopTrace);
}
```

Listing 1. Section of a TTCN-3 implementation for a client association (cf. Table I).

traffic in both directions. However, separating the traffic has the advantage that it is easier to understand the code during the implementation phase and the logging output of the execution.

The function `f_configClientServer` sets up the client and server and initializes the ports, `f_deconfigClientServer` releases the ports and resets the client and server.

After configuration, the server component runs `f_Replay`, a function that simply receives messages from the observation port and forwards them to the client component via the comm port. The client component runs the function `f_Associate`. It takes two arguments, the `requestPDU` is the association request message that has to be sent by the client and the template `responsePDU` to match the expected response. Internally, it also uses `negativeResponsePdu`, a globally defined template for negative responses. The function first starts the sniffer by sending the command `startTrace` via the observation port and triggers the association by sending the request message via the control port. Subsequently, it waits until it receives a message that matches the request PDU on the observation port and a message that matches the response PDU template or the negative response PDU on the comm port. Any other 61850 messages are simply ignored.

The test is successful (the verdict is *pass*) if matching messages are received in the correct order; the verdict is *fail* if the second message is a negative response. Otherwise the test will remain in one of the `alt`-statements. Finally the function `f_waitAndGuard` implements a timeout. It waits until either the client component is done with the execution of `f_Associate` or a predefined timeout is reached. If that happens it terminates the test case and the verdict is set to *inconclusive*.

It is very easy to extend this setup for most, if not all, other client tests that do not require any direct interaction between the test system and the IEC 61850 server, i.e., test cases for ‘normal’ operation. We have in fact successfully implemented several test cases including unbuffered and buffered reporting and direct as well as select-before-operate control models.

Testing the client’s behaviour in irregular circumstances requires a non-conformant server. As our server implementation was part of a (conformance certified) SCADA system we were unable to use it for such test cases. The alternative is using a (mock-up) server implementation in TTCN-3 that simply sends (preconfigured) messages to emulate certain behaviour. However, implementing such a server was beyond the scope of our experiments.

B. IEC 61850-90-5 synchrophasor performance tests

As we were convinced that TTCN-3 can be used for all kinds of client and server interoperability tests we extended our investigations towards performance tests for the transmission of synchrophasors according to the technical report TR IEC 61850-90-5 [14].

Synchrophasors are measured by Phasor Measurement Units (PMU). A PMU is a device that measures voltage magnitude and phase angle and current magnitude and phase angle relative to a known time-reference. Additionally, the frequency and frequency drift (rate of change of frequency (ROCOF)) are estimated. Compared to conventional remote terminal units much higher sampling rates of up to 50/60 per second are supported (sometimes even up to 120/s). PMUs are a key sensor to establish a modern wide area monitoring system and they enable a multitude of energy applications [15][16].

We selected the use case ‘Under voltage load shedding’ (Section 5.9.4 of [14]): An intelligent electronic device (IED) receives synchrophasor measurements from multiple PMUs. It detects a voltage collapse by observing an unusual continuous voltage deviation and generates a GOOSE event that instructs a circuit breaker to trip a line. All communication delays must not exceed 100ms and the synchrophasor measurement timing error must be less than 50µs.

Our test case setup can be seen in Figure 3. PMU 1 and the PMU tester are installed in Salzburg while the rest is located in a testbed in Berlin. Obviously the PMU tester is used to generate unusual voltage deviations and the WAN emulator emulates a WAN connection by adding delay, jitter, packet loss and bit errors. Additionally, GPS is used as a very precise time source at each site.

We developed the following test cases:

- Communication delay: For varying emulated network delays (0-100ms) the duration from measuring a phasor until it is processed by the IED is measured. A test passes if the duration is below 100ms.

 ISBN: 978-1-61208-406-0

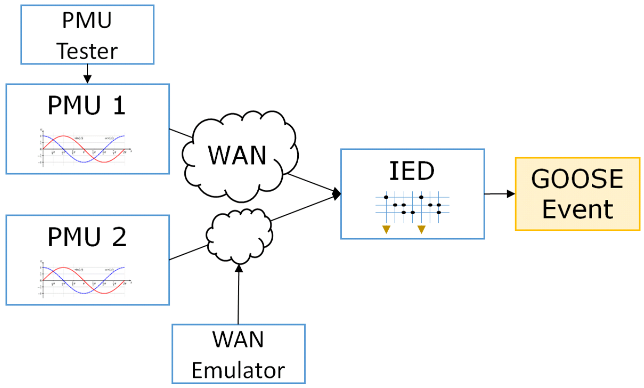

Figure 3. Setup for the IEC 61850 'Under voltage load shedding' performance test case.

- Time to trip: For varying network conditions (delay, loss, bit errors) the time between a significant voltage deviation and the GOOSE event is measured. A test passes if the GOOSE message is sent within a predefined threshold.

These test cases are much more complex than the previously conducted client tests, and so is the test setup. Naturally each device that has to be controlled during the test needs a system adapter. But the biggest difference is clearly that these devices are running at two separate sites. Fortunately TTCN-3 can also run distributed tests.

A distributed TTCN-3 test system consist of one test management (TM) and component handling (CH) instance at the master test component (MTC) and a number of parallel test components (PTC), each with it's own test executable, codec, system and platform adapters. The general structure can be seen in Figure 1. However, TTCN-3 handles the distribution and from an implementers point of view there is no difference between local and remote test components.

V. Conclusion and Future Work

Abstract test cases in TTCN-3 are much more formal than the currently used prose descriptions, while still being easy to understand without extensive training as a TTCN-3 developer. Special language constructs like the `alt`-statement or the powerful templating and data matching features of TTCN-3 offer huge benefits over general purpose programming languages.

By using TTCN-3 we were able to define abstract test cases for conformance testing of an IEC 61850 client, import the data types from an ASN.1 definition and execute these test cases in a TTCN-3 runtime environment to test a concrete client against a certified server. By implementing proper initialization and cleanup functionality in the system adapters a whole test suite can be executed automatically without any human intervention. Furthermore we were able to define performance tests for IEC 61850-90-5 synchrophasor transmissions where the SUT is distributed over two sites.

Our next goals are the definition and implementation of several other test cases for synchrophasor transmission, including Phasor Data Concentrators (PDC) and a Data Archiver (DA). We will also investigate how TTCN-3 can be used to test security mechanisms, e.g., key updates from a key distribution center.

Acknowledgment

This work is partially funded by FFG (Österreichisches Forschungsförderungsgesellschaft) through the ERA-SME project OTITOS.

References

[1] K. C. Budka, J. G. Deshpande, and M. Thottan, Communication Networks for Smart Grids: Making Smart Grid Real, 1st ed., ser. Computer Communications and Networks. Springer, London, England, 2014.

[2] Industrial Automation systems - Manufacturing Message Specification - Part 1: Service Definition, ISO Std. ISO/IEC 9506-1:2003, 2003.

[3] Industrial Automation systems - Manufacturing Message Specification - Part 2: Protocol Specification, ISO Std. ISO/IEC 9506-2:2003, 2003.

[4] "UCA International User Group," [Online]. Available from: http://www.ucaiug.org/aboutUCAIug/default.aspx, 2015.03.10.

[5] Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Part 1: General concepts, ISO Std. ISO/IEC 9646-1:1994, 1994.

[6] Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Part 3: The Tree and Tabular Combined Notation (TTCN), ISO Std. ISO/IEC 9646-3:1994, 1994.

[7] Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 1: TTCN-3 Core Language, ETSI Std. ETSI ES 201 873-1 V4.6.1, 2014.

[8] "TTCN-3 Standardized ETSI Test Suites," [Online]. Available from: http://www.ttcn-3.org/index.php/downloads/publicts/publicts-etsi, 2015.03.10.

[9] "TTCN-3 Standardized 3GPP Test Suites," [Online]. Available from: http://www.ttcn-3.org/index.php/downloads/publicts/publicts-3gpp, 2015.03.10.

[10] Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 5: TTCN-3 Runtime Interface (TRI), ETSI Std. ETSI ES 201 873-5 V4.6.1, 2014.

[11] S. Schwabe, Q. H. Nguyen, and T. Vassiliou-Gioles, "Synchronized Distributed Testing using TTCN-3 and GPS," ETSI TTCN-3 User Conference 2007, Stockholm, Sweden, 2007.

[12] S. Gröning, C. Lewandowski, J. Schmutzler, and C. Wietfeld, "Interoperability Testing based on TTCN-3 for V2G Communication Interfaces," in Proceedings of the International Conference on Connected Vehicles and Expo (ICCVE), Beijing, China, 2012, pp. 298–303.

[13] M. Flohil and R. Schimmel, "Conformance Test Procedures for Client System with IEC 61850-8-1 interface. Revision 1.1," KEMA Consulting, 2009.

[14] Communication networks and systems for power utility automation – Part 90-5: Use of IEC 61850 to transmit synchrophasor information according to IEEE C37.118, IEC Std. IEC 61 850-90-5, 2012.

[15] M. Gavrilas, "Recent Advances and Applications of Synchronized Phasor Measurements in Power Systems," in Proceedings of the 9th WSEAS/IASME International Conference on Electric Power Systems, High Voltages, Electric Machines, Budapest, Hungary, 2009.

[16] M. Patel et al., "Real-Time Application of Synchrophasors for Improving Reliability," North American Electric Reliability Corporation, Princeton, NJ, USA, 2010.

 ISBN: 978-1-61208-406-0