INL/CON-12-25647
PREPRINT

Improving Cyber-Security of Smart Grid Systems Via Anomaly Detection and Linguistic Domain Knowledge

5th International Symposium on Resilient Control Systems

Ondrej Linda
Milos Manic
Todd Vollmer

August 2012

The INL is a
U.S. Department of Energy
National Laboratory
operated by
Battelle Energy Alliance

Improving Cyber-Security of Smart Grid Systems via Anomaly Detection and Linguistic Domain Knowledge

Ondrej Linda, Milos Manic
University of Idaho
Idaho Falls, ID, USA
olinda@uidaho.edu, misko@ieee.org

Todd Vollmer
Idaho National Laboratory
Idaho Falls, ID, USA
denis.vollmer@inl.gov

***Abstract*— The planned large scale deployment of smart grid network devices will generate a large amount of information exchanged over various types of communication networks. The implementation of these critical systems will require appropriate cyber-security measures. A network anomaly detection solution is considered in this paper. In common network architectures multiple communications streams are simultaneously present, making it difficult to build an anomaly detection solution for the entire system. In addition, common anomaly detection algorithms require specification of a sensitivity threshold, which inevitably leads to a tradeoff between false positives and false negatives rates. In order to alleviate these issues, this paper proposes a novel anomaly detection architecture. The designed system applies a previously developed network security cyber-sensor method to individual selected communication streams allowing for learning accurate normal network behavior models. In addition, an Interval Type-2 Fuzzy Logic System (IT2 FLS) is used to model human background knowledge about the network system and to dynamically adjust the sensitivity threshold of the anomaly detection algorithms. The IT2 FLS was used to model the linguistic uncertainty in describing the relationship between various network communication attributes and the possibility of a cyber attack. The proposed method was tested on an experimental smart grid system demonstrating enhanced cyber-security.**

Keywords*— *Anomaly Detection; Critical Systems, Cyber Sensor; Fuzzy Logic System; Domain Knowledge; Smart Grid;

I. INTRODUCTION

Resiliency and enhanced state-awareness are highly desirable properties of modern critical systems [1]. It is of paramount importance that critical infrastructures, such as energy production or energy distribution systems, are equipped with intelligent components for timely reporting and understanding of the status and behavioral trends in the system [2]. With the increasing amount of information being exchanged over various types of communication networks, resiliency and enhanced state-awareness cannot be achieved without ensuring appropriate cyber-security measures.

In the particular case of smart grids networks a large scale deployment of devices will soon be prevalent. These systems potentially add Wireless Access Point (WAP) devices to existing utility networks. For instance, in a typical Advanced Metering Infrastructure (AMI) system 1,500 wireless sensors report to one or multiple WAP nodes [3]. As of April 2010, almost 69 million of these meters were planned for deployment in the United States [4]. Assuming a uniform deployment of sensors this calls for 46,000 WAP's without any regard for redundancy. An example deployment is the Pacific Northwest Smart Grid Demonstration Project. A 2011 progress report states that utility partners are in the process of installing 80,000 smart grid components to consumers in five states [5]. This large influx of devices into a network vastly expands the potential network attack surface.

To ensure the cyber-security of network system various approaches can be applied [6]-[14]. One of the most common approaches is anomaly detection. An anomaly detection system is trained on a set of normal network behavior. The extracted behavior model is then used to detect anomalous behavior in the newly observed testing data.

Two possible difficulties with this approach are identified as follows. Firstly, building a single comprehensive normal behavior model for a specific network communication system might be difficult due to the complexity of the network and due to the presence of multiple diverse communication streams. Secondly, the performance of anomaly detection algorithms can be tuned by adjusting a sensitivity threshold. The selection of a specific threshold value inevitably results in a tradeoff between false negative and false positive rate. Hence, determining the suitable sensitivity threshold value constitutes an important design problem.

This paper alleviates the above mentioned issues by proposing novel anomaly detection architecture. The presented system first identifies individual communication streams in the overall network traffic and then individually applies a previously developed network security cyber-sensor algorithm to selected streams [8], [15]. This approach allows for learning accurate normal behavior models specific to each network communication. In addition, an Interval Type-2 Fuzzy Logic System (IT2 FLS) is used to model human background knowledge about the network system and to dynamically adjust the sensitivity threshold of the anomaly detection algorithms. The IT2 FLS is used to model the linguistic uncertainty in describing the relationship between various network communication attributes and the possibility of a cyber attack. For instance, if only a small number of distinct communication protocols is expected to be used during the

normal network communication, a linguistic rule can be created that sets a lower sensitivity threshold when a high number of distinct communication protocols appear in the network communication. Hence, the IT2 FLS is not used directly for detecting anomalous network traffic, but it is only used to utilize the provided human domain knowledge to tune the performance of the clustering based anomaly detection algorithm via adjusting the sensitivity threshold.

The proposed anomaly detection system was implemented and tested on a smart grid experimental test-bed. It was demonstrated that the system can learn normal behavior models for each selected communication stream and perform accurate anomaly detection. In addition, it was also demonstrated that the availability of domain knowledge can significantly improve the performance of the anomaly detection method.

The rest of the paper is structured as follows. Section II presents an overview of the previously developed network security cyber-sensor. Section III proposes how to model the domain knowledge using IT2 FL rules. Section IV describes the architecture of the proposed anomaly detection system. Finally, the system is tested in Section V and the paper is concluded in Section VI.

II. Previous Work

This section provides a brief overview of the previously developed network security cyber-sensor algorithm. First, the network traffic feature extraction method is described. Next, the fuzzy rule extraction technique based on online clustering is explained.

A. Feature Extraction from Packet Stream

The anomaly detection algorithm is trained on a set of network traffic features extracted by a window-based technique. This technique is applied directly to the stream of packets. The inherent time series nature of the packet stream data is described by a vector, which captures statistical properties of the network traffic.

As described in the previous work [8], a window of specified length is shifted over the stream of network packets. At each position of the window a descriptive feature vector is computed. As the next arriving packet is pushed into the window, the last packet is removed from the end. Fig. 1 schematically depicts this feature extraction process. Table I summarizes the list of extracted statistical features from the packet window. This set of features was empirically selected based on the motivation to most accurately capture the time series nature of the packet stream. For further details and evaluation of the feature extraction refer to [8].

B. Fuzzy Logic Rule Extraction via Online Clustering

In the previous work of the authors, a new low-cost online rule extraction technique was proposed to model the network traffic [8]. The model is composed of a set of fuzzy rules that are constructed based on the window-based feature vectors using an online version of the adapted Nearest Neighbor Clustering (NNC) algorithm. This adapted algorithm

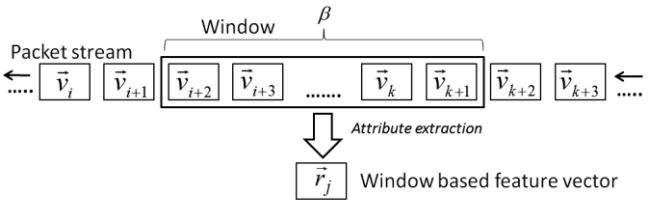


Fig. 1 Window based feature extraction process [10].

TABLE I
Selected Window-Based Features

Num. of IP addresses	Num. of Flag Codes
Min. Num. of Packets / IP	Min. Num. of Packets / Flag Code
Max. Num. of Packets / IP	Max. Num. of Packets / Flag Code
Avg. Time between Packets	Num. of Packets with 0 Win. Size
Time Length of the Window	Num. of Packets with 0 Data Len.
Data Speed	Avg. Win. Size
Num. of Protocols	Avg. Data Length
Min. Num. of Packets / Protocol	Num. of Ports
Max. Num. of Packets / Protocol	

maintains additional information about the spread of data points associated with each cluster throughout the clustering process. Each cluster P_i of encountered normal network behavior is described by its center of gravity $\vec{c}_i$, weight w_i and a matrix of boundary parameters M_i. Hence:

$$P_i = \{\vec{c}_i, w_i, M_i\},\ \vec{c}_i = \{c_i^1, \ldots, c_i^n\},\ M_i = \begin{vmatrix} c_{i,1}^U & \cdots & c_{i,n}^U \\ c_{i,1}^L & \cdots & c_{i,n}^L \end{vmatrix} \quad (1)$$

Here, i is the index of the particular cluster, c_i^j is the attribute value in the j^{th} dimension, $c_{i,j}^U$ and $c_{i,j}^L$ are the upper and lower bounds of the encountered values of the j^{th} attribute for data points assigned to cluster P_i and n denotes the dimensionality of the input. The algorithm is initialized with a single cluster P_1 positioned at the first supplied training input vector $\vec{x}_1$. This initial input vector is received once the shifting window is first filled with the incoming network packets.

Upon acquiring a new data vector $\vec{x}_i$ from the shifting window buffer, the set of clusters is updated according to the NNC algorithm. First, the Euclidean distance to all available clusters with respect to the new input feature vector $\vec{x}_i$ is calculated. The nearest cluster P_a is identified. If the computed nearest distance is greater than the established maximum cluster radius parameter, a new cluster is created. Otherwise the nearest cluster P_a is updated according to:

$$\vec{c}_a = \frac{w_a \vec{c}_a + \vec{x}_i}{w_a + 1},\ w_a = w_a + 1 \quad (2)$$

$$c_{i,j}^U = \max(x_i^j, c_{i,j}^U),\ c_{i,j}^L = \min(x_i^j, c_{i,j}^L) \quad j = 1 \ldots n \quad (3)$$

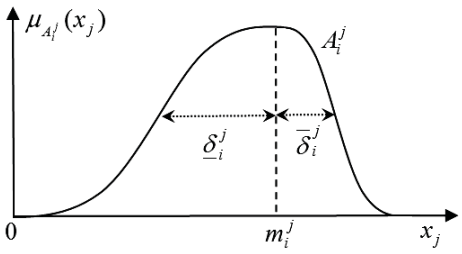


Fig. 2 Illustration of the non-symmetric input Gaussian fuzzy set A_i^j.

The rule extraction phase of the learning process produces a set of clusters, which describe the normal network communication behavior. In the next stage, each cluster is converted into a fuzzy logic rule. Each fuzzy rule describes the belonging of a particular sub-region of the multi-dimensional input space to the class of normal behavior.

Each cluster is transformed into a fuzzy rule. Each fuzzy rule is composed of n antecedent fuzzy sets A_i^j that are modeled using a non-symmetric Gaussian fuzzy membership function with distinct left and right standard deviations. There are three parameters of the membership function, the mean m_i^j and the left and the right standard deviations $\bar{\delta}_i^j$, $\underline{\delta}_i^j$, as shown in Fig. 2. The parameter values are extracted based on the computed cluster P_i in the following manner:

$$m_i^j = c_i^j \tag{4}$$

$$\bar{\delta}_i^j = \alpha(\bar{c}_i^j - c_i^j) \tag{5}$$

$$\underline{\delta}_i^j = \alpha(c_i^j - \underline{c}_i^j) \tag{6}$$

Here, symbol α denotes the fuzziness parameter, which is used to adjust the spread of the membership functions. This set of fuzzy rules is then used to calculate a similarity score between the input vector and the model of normal behavior

C. Anomaly Detection Example

The presented fuzzy logic based anomaly detection method assigns a real value to each window-based feature vector. This value expresses the likelihood that the window of packets contains an intrusion. The closer this value is to 1 the more confident the algorithm is that there is an intrusion present. The classification performance of this anomaly detection algorithm can be tuned by setting a specific sensitivity threshold θ. This threshold adjusts the tradeoff between the false negative and false positive rate of the algorithm.

As an exemplary case study, consider an illustrative output of the presented anomaly detection algorithm as depicted in Fig. 3. Here, the thin solid black line depicts the real-valued response of the anomaly detection algorithm, the thick solid red line marks the actual occurrence of an intrusion and finally the thin dotted line depicts three different sensitivity threshold levels. The classification performance in terms of correct classification rates and the false positive and false negative rates for three different constant sensitivity threshold values is summarized in Table II. It can be observed that lowering the threshold value decreases the false negative rate (i.e. frequency of missed intrusion attempts), however, with the tradeoff of increasing the false positive rate (i.e. frequency of falsely reported alarms).

III. Representating Domain Knowledge Using Linguistic Fuzzy Rules

This Section first provides a brief introduction to Interval Type-2 Fuzzy Logic. Next, the methodology for representing cyber-security domain knowledge is described.

A. Interval Type-2 Fuzzy Logic Systems

Type-1 Fuzzy Sets (T1 FSs) and T1 Fuzzy Logic Systems (FLSs) have been successfully applied in many engineering areas [16]-[18]. However, when modeling linguistic terms, which can mean different things to different people, T1 FSs have been shown to provide only limited design capabilities [18]. To address these issues, Type-2 (T2) FSs and T2 FLSs were originally proposed by Zadeh [19]. T2 FSs offer more modeling flexibility because they employ membership degrees that are themselves fuzzy sets [20]-[22].

In this paper, the Interval T2 (IT2) FSs are considered. IT2 FSs restrict all membership grades into intervals, which result in significant simplification of the computational complexity associated with computing with IT2 FSs. An IT2 FS $\tilde{A}$ can be described by its membership function $\mu_{\tilde{A}}(x,u)$, where $x \in X$ and $u \in J_x$ [18]:

TABLE II
Classification Performance with Different Sensitivity Thresholds

Threshold	Correct Rate	False Pos.	False Neg.
0.3	99.9037%	0.1217%	0.0275%
0.6	99.5504%	0.1082%	1.3753%
0.9	99.3799%	0.1082%	2.0079%

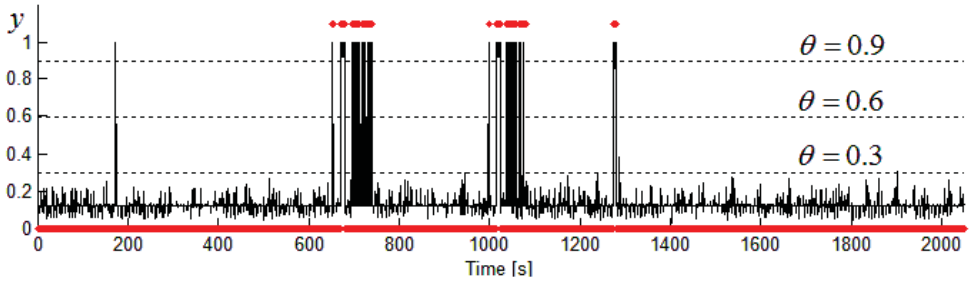


Fig. 3 Classification performance of the fuzzy logic based anomaly detection system with different levels of constant sensitivity threshold θ.

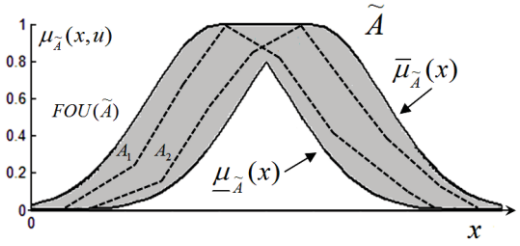


Fig. 4 Interval type-2 fuzzy set $\tilde{A}$.

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} 1/(x,u) \qquad J_x \subseteq [0,1] \tag{7}$$

Here, x and u are the primary and the secondary variables and J_x denotes the interval support of the secondary membership function. The domain of the primary memberships J_x defines the Footprint-Of-Uncertainty (FOU) of FS $\tilde{A}$:

$$FOU(\tilde{A}) = \bigcup_{x \in X} J_x \tag{8}$$

The FOU of an IT2 FS can be completely described by the upper and lower membership functions:

$$FOU(\tilde{A}) = \bigcup_{\forall x \in X} (\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)) \tag{9}$$

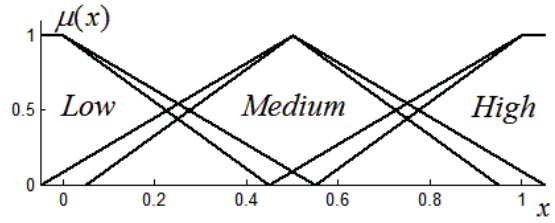
It is this FOU that allows for modeling of linguistic uncertainty. As an example depicted in Fig. 4, consider two possibilities for modeling an arbitrary linguistic concept using T1 FSs A_1 and A_2 (e.g. two experts designed two different membership functions for the same concept) and the possible model of this concepts using IT2 FSs $\tilde{A}$. It can be seen that the IT2 FS encapsulates the T1 FS models and it can model the linguistic uncertainty. This flexibility in modeling vague linguistic concepts was the reason for employing IT2 FSs and IT2 FLS for modeling the linguistic human cyber-security domain knowledge in the proposed system.

Linguistic knowledge can be formulated using implicative IT2 fuzzy rules as follow [18]:

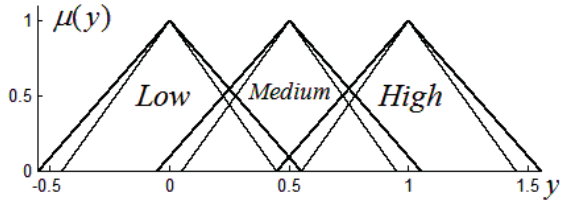
Rule R_k: **IF** x_1 is $\tilde{A}_1^k$ **AND** … **AND** x_n is $\tilde{A}_n^k$
THEN y_k is $\tilde{B}^k$ (10)

Here, symbols $\tilde{A}_i^k$ and $\tilde{B}^k$ denote the i^{th} input IT2 FS and the output IT2 FS of the k^{th} rule, respectively, where n is the dimensionality of the input vector $\vec{x}$ and y_k is the associated output variable.

The set of linguistic rules together with the representation of the input and output IT2 FSs can be used to create an IT2 FLS. Due to the limited space in this paper, the technical details of fuzzy inferencing using IT2 FLSs have been omitted but they can be found in literature [18], [23].



(a)



(b)

Fig. 5 Input IT2 FSs (a) and output IT2 FSs (b).

B. Cyber-Security Domain Knowledge Modeling

The IT2 fuzzy rules can be used to linguistically describe the relationship between various features of the network communication and the possibility of a cyber attack. The window-based feature extraction technique is used to describe the global features of the monitored network traffic.

Each window-based feature is first normalized into a unit interval. There are different approaches to fuzzifying the input domain of each attribute. Because of its simplicity, the fuzzification scheme depicted in Fig. 5(a) was used in the presented work. Here, two trapezoidal and one triangular IT2 fuzzy sets were used to fuzzify each input domain into fuzzy sets "*Low*", "*Medium*" and "*High*".

The output IT2 FSs express the likelihood of an intrusion in the system and can be used to adjust the sensitivity threshold of each anomaly detection algorithm. As was chosen for the input domain, the output domain is modeled using the three triangular IT2 FSs: "*Low*", "*Medium*" and "*High*". These sets are depicted in Fig. 5(b).

The provided set of linguistic fuzzy rules and the described input and output IT2 FSs are used to implement an IT2 FLSs, which calculates the specific sensitivity threshold of the anomaly detection. For instance, the domain knowledge can be encoded using IT2 FL rules such as: "*If number of protocols is high then sensitivity threshold is low*".

IV. Anomaly Detection System Using Linguistic Rules

The overall architecture of the proposed anomaly detection system is depicted in Fig. 6. The network traffic is first processed by an IT2 FLS which uses a fuzzy logic rule base with encoded linguistic domain knowledge to calculate the cyber-security context of the current observed network traffic. This cyber-security context expresses the belief that an intruder is currently present in the system.

In the next stage, the network traffic is separated into individual communication streams. In the current implementation, a specific IP address is used to identify each communication stream. Other features, such as port numbers

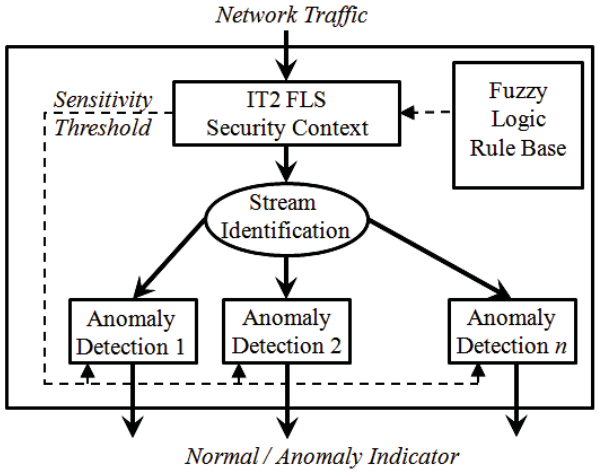


Fig. 6 Architecture of the proposed anomaly detection system.

of protocol types can also be used. Packets assigned to individual communication streams are then passed into dedicated anomaly detection algorithms. Each anomaly detection algorithm maintains its own buffer of incoming packets, which is used to extract the window-based features as described in Section II. The fuzzy logic based anomaly detection algorithm is used to assign a real value to each input vector, which expresses the belief that the current packet window contains intrusive packets. The closer this value is to 1 the more confident the algorithm is that an intrusion is present.

The final classification is performed by comparing the real-valued output to the sensitivity threshold. When the real-valued output is above the sensitivity threshold, a network anomaly is reported for the specific communication stream. When the output value is below the sensitivity threshold the network traffic is marked as normal. The actual value of the sensitivity threshold is dynamically computed based on the cyber-security context computed by the IT2 FLS. Hence, the IT2 FLS encoding human domain knowledge is not used directly for detecting anomalies, instead it is used to only tune the performance of the anomaly detection algorithm via adjusting the sensitivity threshold.

It should be noted here that the anomaly detection algorithm utilizes an assumption that a representative normal behavior training data set has been collected. In case, that a representative normal behavior training data set was not collected, the anomalous classification of the network traffic might only signalize that the observed network traffic is normal but it has not been included in the training data set. This assumption is a fundamental concept underlying the use of anomaly detection techniques.

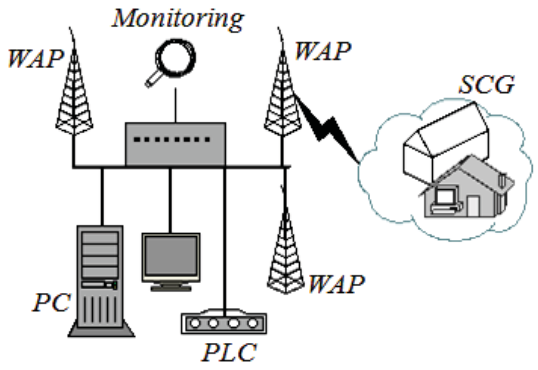


Fig. 7 Diagram of the smart grid experimental test-bed.

V. Experimental Results

This Section first describes the smart grid experimental test-bed and then presents experimental results.

A. Experimental Test-Bed

A small campus grid (SCG) and sensor network that physically exists in the Center for Advance Energy Studies in Idaho Falls, Idaho was used as a smart grid test platform. The network consists of a heterogeneous mixture of devices including wireless sensors monitoring environmental conditions in the building, wind and solar renewable resources, and a variety of control system devices. The SCG is connected to a small wind turbine, a solar power station, and a wireless AMI infrastructure with two WAP's. A representation of the sensor network and small campus grid is shown in Fig. 7. Additionally, the network has several Windows based computers, web camera's, a Rockwell Automation PLC and a National Instruments PLC.

The SCG includes a wireless sensor network consisting of environmental sensors from three commercial vendor systems. The network contains wireless systems from Emerson, Honeywell and Arch Rock. Each system connects wirelessly to the sensors via a wireless access point. As with the AMI deployment these WAP gateways have a wired connection on one side of the network and wireless interfaces to the remote sensors on the other side. The network capture device has visibility on the wired side of the connection. Each wired WAP connection varies in the method of network protocols utilized on top of Ethernet.

B. Experimental Results

In order to obtain suitable testing data, the *Nmap* [24] and *Nessus* [25] software applications were used to generate anomalous network traffic behavior in an attempt to simulate instances of cyber attacks. *Nmap* is a network scanning tool that is commonly used to identify hosts, scan ports, operating systems and to determine applications that are listening on open ports. *Nessus* provides auditing capabilities, vulnerability assessments and profiling information. The simulated intrusion attempts included: ARP pings, SYN stealth scans, port scanning, open port identification and others. Cyber attacks ranged from long attacks composed of many packets to very short intrusion sequences.

Training and testing datasets of experimental network traffic were recorded. The training data set contained 100,000 packets recorded during normal network activity. Here, the normal network activity refers to a common network communication traffic flow without any disturbances. In order to obtain this normal training data set, isolated network traffic was maintained to prevent the possibility of the presence of any intrusive attempts. This data set was used only during the

TABLE III
CYBER-SECURITY LINGUISTIC DOMAIN KNOWLEDGE

R_1:	If **Time of Window** is *Low* then **Sensitivity Threshold** is *Low*
R_2:	If **Time of Window** is *Medium* then **Sensitivity Threshold** is *Low*
R_3:	If **Time of Window** is *High* then **Sensitivity Threshold** is *High*
R_4:	If **Number of Protocols** is *Low* then **Sensitivity Threshold** is *High*
R_5:	If **Number of Protocols** is *Medium* then **Sensitivity Threshold** is *High*
R_6:	If **Number of Protocols** is *High* then **Sensitivity Threshold** is *Low*

training phase of the algorithms. The second data set contained 200,000 recorded packets with simulated abnormal behavior along with normal behavior. This data set was not used during the training phase.

For this specific experimental test bed, a set of six linguistic fuzzy rules was used to summarize the domain knowledge as shown in Table III. The first three rules were derived from the knowledge that the expected normal network traffic features steady behavior with only minor variations in the rate of transmitted packets. The second three rules then express the knowledge that the present system uses only a small number of communication protocols and an increased number of different communication protocols are a likely indication of possible intrusive attempt.

With three selected communication streams, the training phase took 4.03s seconds of wall clock time while testing was achieved in 15.12s. The fuzzy logic models for individual communication streams were composed of 19, 57 and 2 clusters, respectively. Fig. 8 depicts the results of the anomaly detection for the three selected communication streams. The dotted line depicts the dynamically calculated sensitivity threshold. It can be observed that the provided linguistic domain knowledge encoded in form of IT2 fuzzy rules allows for dynamic adjustment of the sensitivity threshold.

The classification performance of the proposed anomaly detection system is compared to the classification performance with constant sensitivity threshold in Tables IV-VI. It can be observed that the proposed method achieves the best tradeoff between the rate of false positives and false negatives. In other words, the experimental results demonstrate that when relevant domain knowledge about the specific network system is available, it can be utilized to improve the classification performance of the network anomaly detection method via dynamically adjusting the sensitivity threshold.

VI. CONCLUSION

This paper presented a novel complex anomaly detection architecture for critical control systems. The proposed system applied a previously developed network security cyber-sensor method to individual selected communication streams. In

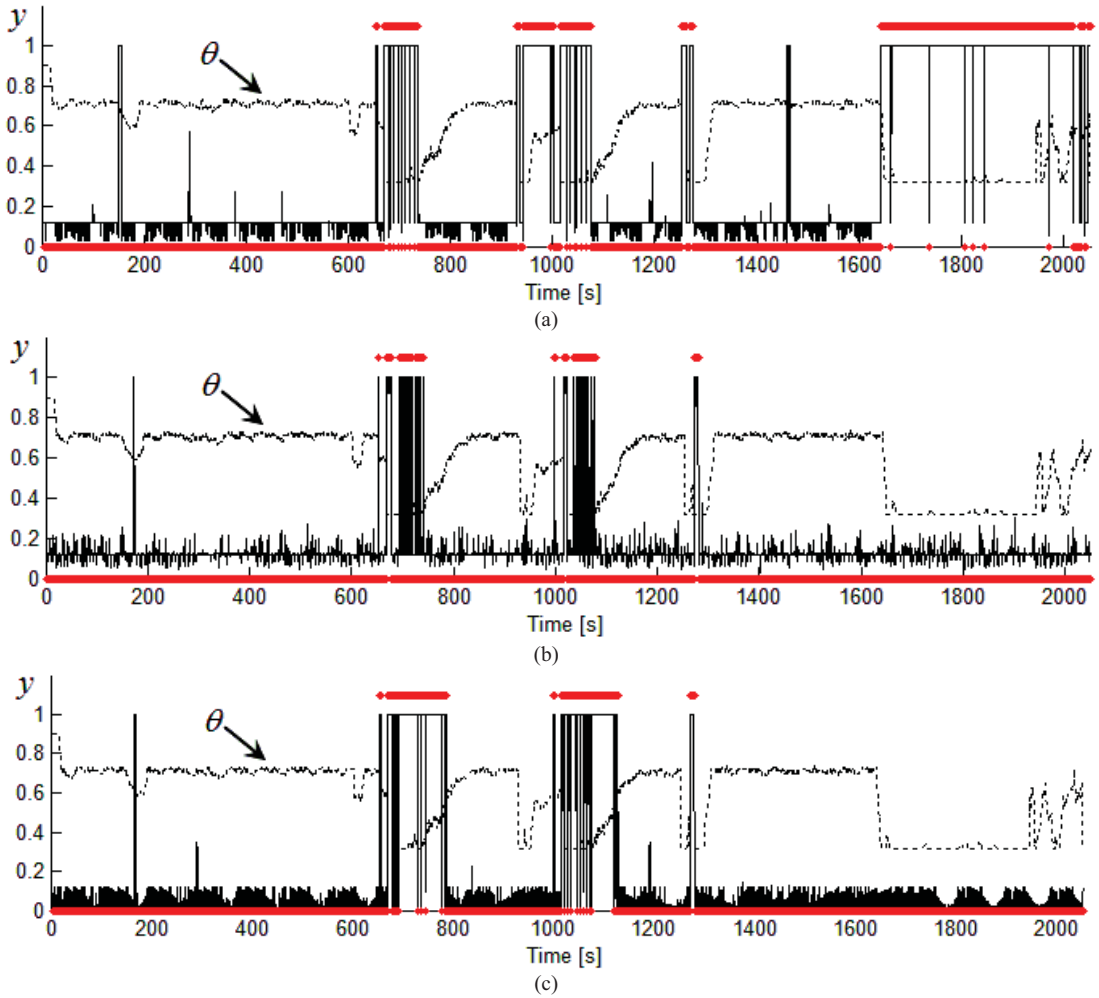


Fig. 8 Classification performance of the proposed anomaly detection system for three selected communication streams (a)-(c).

TABLE IV
CLASSIFICATION PERFORMANCE FOR STREAM 1

Threshold	Correct Rate	False Pos.	False Neg.
0.3	99.8539%	0.1461%	0.0000%
0.6	99.8705%	0.1295%	0.0000%
0.9	99.8788%	0.1212%	0.0000%
IT2 FLS	99.8722%	0.1278%	0.0000%

TABLE V
CLASSIFICATION PERFORMANCE FOR STREAM 2

Threshold	Correct Rate	False Pos.	False Neg.
0.3	99.9037%	0.1217%	0.0275%
0.6	99.5504%	0.1082%	1.3753%
0.9	99.3799%	0.1082%	2.0079%
IT2 FLS	99.9111%	0.1116%	0.0275%

TABLE VI
CLASSIFICATION PERFORMANCE FOR STREAM 3

Threshold	Correct Rate	False Pos.	False Neg.
0.3	99.8643%	0.2953%	0.0000%
0.6	99.8960%	0.2265%	0.0000%
0.9	99.8960%	0.2265%	0.0000%
IT2 FLS	99.8960%	0.2265%	0.0000%

addition, the developed system dynamically adjusts the sensitivity threshold of each anomaly detection algorithm based on domain knowledge about the specific network system. This domain knowledge was encoded using Interval Type-2 Fuzzy Logic rules, which linguistically describe the relationship between various features of the network communication and the possibility of a cyber attack.

The proposed anomaly detection system was implemented and tested on a smart-grid experimental test-bed. It was demonstrated that the system can learn normal behavior models for individual selected communication streams and perform accurate anomaly detection. In addition, it was also demonstrated that the availability of domain knowledge can significantly improve the performance of the anomaly detection method by dynamically adjusting the sensitivity threshold.

ACKNOWLEDGMENT

The authors acknowledge support for this work from Idaho National Laboratory through the U.S. Department of Energy Office of Electrical Delivery and Energy Reliability under DOE Idaho Operations Office Contract DE-AC07-05ID14517.

REFERENCES

[1] C. G. Rieger, D. I. Gertman, M. A. McQueen, "Resilient Control Systems: Next Generation Design Research," in *Proc. 2nd IEEE Conf. on Human System Interactions*, pp. 632-636, May 2009.

[2] D. Yang, A. Usynin, J. W. Hines, "Anomaly-Based Intrusion Detection for SCADA Systems," in *Proc. of 5th Intl. Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies (NPIC&HMIT 05)* , Nov 12-16, 2006.

[3] T. Iwao, K. Yamada, M. Yura, Y. Nakaya, A. Cardenas, S. Lee and R. Masuoka, "Dynamic Data Forwarding in Wireless Mesh Networks," in *Proc. IEEE SmartGridComm Conf.*, pp. 385-390, Oct. 2010.

[4] Utility-Scale Smart Meter Deployments, Plans & Proposals, The Edison Foundation, [URL], Available: http://www.edisonfoundation.net, from April 2012.

[5] Pacific Northwest Smart Grid Demonstration Project – 2011 Annual Report [URL], Available: http://www.pnwsmartgrid.org/publications.asp, from April 2012.

[6] F. Gonzalez, D. Dasgupta, J. Gomez, M. Kaniganti, "An Evolutionary Approach to Generate Fuzzy Anomaly Signatures," in *Proc. the IEEE Information Assurance Workshop*, pp. 251-259, June 2003.

[7] J. Gomez, D. Dasgupta, F. Gonzalez, "Detecting Cyber Attacks with Fuzzy Data Mining Techniques," in *Proc. of the Workshop on Data Mining for Counter Terrorism and Security, 3rd SIAM Conference on Data Mining*, May 2003.

[8] O. Linda, T. Vollmer, J. Wright, M. Manic, "Fuzzy Logic Based Anomaly Detection for Embedded Network Security Cyber Sensor," in *Proc. IEEE Symposium Series on Computational Intelligence*, pp. 202-209, April 2011.

[9] Z. Zhang, J. Li, C. Manikopulos, J. Jorgenson, J. Ucles, "HIDE: a Hierarchical Network Intrusion Detection System Using Statistical Preprocessing and Neural Network Classification," in *Proc. IEEE Workshop on Information Assurance and Security*, pp. 85-90, 2001.

[10] O. Linda, T. Vollmer, M. Manic, "Neural Network Based Intrusion Detection System for Critical Infrastructures," in *Proc. Int. Joint INNS-IEEE Conf. on Neural Networks*, pp. 1827-1834, June 14-19, 2009.

[11] W. Hu, Y. Liao, V. R. Vemuri, "Robust Anomaly Detection Using Support Vector Machines," in *Proc. International Conference on Machine Learning*, pp. 592-597, 2003.

[12] G. Stein, B. Chen, A. S. Wu, K. A. Hua, "Decision Tree Classifier For Network Intrusion Detection With GA-based Feature Selection," in *Proc. of the 43rd ACM Southeast Conference*, pp. 136-141, March 2005.

[13] S. Zhong, T. Khoshgoftaar, N. Seliya, "Clustering-based network intrusion detection," in *Intl. Journal of Reliability, Quality and Safety*, Vol. 14, No. 2, pp. 169-187, 2007.

[14] Q. Wang, V. Mehalooikonomou, "A Clustering Algorithm for Intrusion Detection," in *SPIE Conference on Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security*, pp. 1083-1086, 2005.

[15] O. Linda, T. Vollmer, M. Manic, J. Alves-Foss, "Towards Resilient Critical Infrastructures: Application of Type-2 Fuzzy Logic in Embedded Network Security Cyber Sensor," in *Proc. IEEE Symposium on Resilience Control Systems*, pp. 26-32, Aug., 2011.

[16] L. A. Zadeh, "Fuzzy Sets," in *Information and Control*, vol. 8, pp. 338-353, 1965.

[17] J. Valente de Oliveira, W. Pedrycz (eds.), *Advances in Fuzzy Clustering and its Applications*, John Wiley & Sons, Ltd, 2007.

[18] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Upper Saddle River, NJ: Prentice Hall PTR, 2001.

[19] L. A. Zadeh, "The Concept of a Linguistic Variable and its Approximate Reasoning - II," in *Information Sciences*, No. 8, pp. 301-357, 1975.

[20] H. A. Hagras, "A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots," in *IEEE Trans. Fuzzy Systems*, vol. 12, no. 4, pp. 524-539,Aug. 2004.

[21] M. Beglarbegian, W. Melek, J. M. Mendel, "On the robustness of Type-1 and Type-2 fuzzy logic systems in modeling," in *Information Sciences*, vol. 181, issue: 7, pp. 1325-1347, April 2011.

[22] O. Linda, M. Manic, "Interval Type-2 Fuzzy Voter Design for Fault Tolerant Systems," in *Information Sciences*, vol. 181, issue: 14-15, pp. 2933-2950, July 2011.

[23] J. M. Mendel, R. John, F. Liu, "Interval Type-2 Fuzzy Logic Systems Made Simple," in *IEEE Trans. on Fuzzy Systems*, vol. 14, no. 6, pp. 808-821, Dec. 2006.

[24] Nmap webpage [URL], Available: http://nmap.org, from April 2012.

[25] Nessus webpage [URL], Available: http://tenable.com/products/nessus, from April 2012.