Framework for Data Driven Health Monitoring of Cyber-Physical Systems

Kasun Amarasinghe[1], Chathurika Wickramasinghe[1], Daniel Marino[1], Craig Rieger[2], Milos Manic[1]
[1]Virginia Commonwealth University, Richmond, Virginia, USA
[2]Idaho National Laboratory, Idaho Falls, Idaho, USA
amarasinghek@vcu.edu,craig.rieger@inl.gov, misko@ieee.org

***Abstract*— Modern infrastructure is heavily reliant on systems with interconnected computational and physical resources, named Cyber-Physical Systems (CPSs). Hence, building resilient CPSs is a prime need and continuous monitoring of the CPS operational health is essential for improving resilience. This paper presents a framework for calculating and monitoring of health in CPSs using data driven techniques. The main advantages of this data driven methodology is that the ability of leveraging heterogeneous data streams that are available from the CPSs and the ability of performing the monitoring with minimal a priori domain knowledge. The main objective of the framework is to warn the operators of any degradation in cyber, physical or overall health of the CPS. The framework consists of four components: 1) Data acquisition and feature extraction, 2) state identification and real time state estimation, 3) cyber-physical health calculation and 4) operator warning generation. Further, this paper presents an initial implementation of the first three phases of the framework on a CPS testbed involving a Microgrid simulation and a cyber-network which connects the grid with its controller. The feature extraction method and the use of unsupervised learning algorithms are discussed. Experimental results are presented for the first two phases and the results showed that the data reflected different operating states and visualization techniques can be used to extract the relationships in data features.**

Keywords*— *Cyber-Physical Systems; Resilience; Unsupervised learning; Health Monitoring; Explainable AI; Anomaly Detection;

I. INTRODUCTION

Modern systems commonly consist of interconnected computing and physical resources which enable interactive processing among systems [1]. Such systems, called Cyber-Physical Systems (CPSs) integrate computations, communication, control and physical processes to achieve a specific task [2]. Modern infrastructure and systems in many domains have become heavily reliant on CPSs and they can be found in areas ranging from sensor networks [3], intelligent transportation systems and smart grids to space exploration systems [1], [2], [4]–[6]. Due to wide spread usage, ensuring security and resiliency of CPSs is of utmost importance for many socioeconomic reasons.

CPSs can be vulnerable to cyber-attacks and has a great potential of facing security threats without any sign of physical component failure [4]. Further, vulnerability of one individual component can lead to catastrophic cascading failures [2]. Therefore, it is necessary to build resilient CPSs with low failure rates and systems with high recovery rates in a failure [2]. In building resilient CPSs continuous monitoring of the system state is of utmost importance to detect abnormal behavior and diagnose any faulty hardware/software components in real-time [2]. In other words, continuous monitoring of the CPS component and overall health is crucial for improving resilience.

Health monitoring of CPSs can be very complex [4]. The monitoring process must be performed accurately, in real-time. Therefore, the health monitoring process should be able to minimize the time needed to detect and restore the system back to health state [4]. Even though there exist a large body of work for intrusion detection systems in the CPS domain, to the best of our knowledge there's limited work that focus on real time health monitoring of CPSs. In one such effort, Zhang et. al. proposed an adaptive 'health monitoring and management (HMM)' system in order to fulfil the high reliability and safety requirements of CPSs [2]. Their HMM system monitors the health condition of the system, diagnose and identifies the faulty components by implementing a fault signature matrix (FSM). FSM associates the sensors and target system components with the rules which describe the normal behavior of the system. Hackmann et al. proposed a structural health monitoring system (SHM) [7]. In the SHM, they focused on structural deficiencies (environmental corrosion, persistent traffic and wind loading,) that occur during the lifetime of civil infrastructure such as bridges. Lee et al. performed a case study on preparing a predictive health monitoring solution for a fleet of 30 industrial robots by using torque and speed data measurements [8]. Further, they presented an overview of the electric vehicle and battery health management and prognostic platform which is based on several health measurements (stress factors) such as such as environment temperature, humidity, driving style, charging level, discharge rate and road condition.

This paper presents a framework for measuring the operational health of a CPS. In this work, we consider three main areas to characterize the health of a CPS: 1) cyber health, 2) physical health and 3) overall system health. The framework consists of four phases: 1) data acquisition and feature extraction, 2) state learning and estimation, 3) cyber-physical health calculation, and 4) operator warning generation. The first three components have an offline role and an online role. The offline roles entail learning from historical data and the online role entails using the trained models for producing inferences on the live data streams. This paper overviews the complete framework and then provides an initial implementation of the first two components as a proof of concept. The implementation is carried out on a CPS testbed that includes a microgrid simulation and a real-time automation controller communicating with the microgrid. In the current implementation, the focus is on the cyber communication of the CPS.

The rest of the paper is organized as follows; Section II overviews the presented framework and its components, Section III provides the background on machine learning algorithms that are used in the implementation; Section IV presents the current implementation of the framework. Section V presents the experiments and their results and finally Section VI concludes the paper.

II. Framework for Data Driven Health Measurement of Cyber Physical Systems

This section presents the presented framework for health measurement of cyber-physical systems. The presented framework is entirely data driven and assumes minimal prior knowledge of the system dynamics. The framework consists of four main components: 1) data acquisition and feature extraction, 3) state learning and state estimation, 3) health evaluation and 4) operator warning generation. Each component has an offline role and an online processing role. The framework is shown in Fig 1 and Fig 2.

In this section, the description of the framework is tied to the testbed that is used for experimentation. The testbed consists of a Microgrid simulation on a Real Time Digital Simulator (RTDS), a Real Time Automation Controller (RTAC), a Data historian for storing the Microgrid state and controller decision (historian) and a cyber-network connecting the components. Fig 3 depicts the connectivity of the testbed.

A. Data Acquisition and Feature Extraction

Data acquisition involves collecting cyber data and physical data. DNP3 communication encapsulated in TCP is used as the cyber communication protocol. As mentioned, the communication between the controller and the historian is considered as cyber data for this work. Packet streams are captured using a packet sniffer and dissected to extract features [16]. For physical data, the power system information is collected through the RSCAD software for the RTDS. Further, the control decisions is stored from the RTAC for each time step as well. The cyber and physical data need to be correlated with timestamps taking the latency of communication into account.

One the raw data are captured, features need to be extracted from the raw data so that machine learning algorithms can extract patterns from the data. Cyber packet stream is processed to extract features which can potentially indicate cyber threats and as a result characterize the cyber health. Similarly, features need to be extracted from the raw Physical data. Feature extraction can be done using domain expert knowledge and using data driven techniques.

B. State Identification and State Estimation

Once the relevant features are extracted, data driven techniques can be used to identify the different states that the

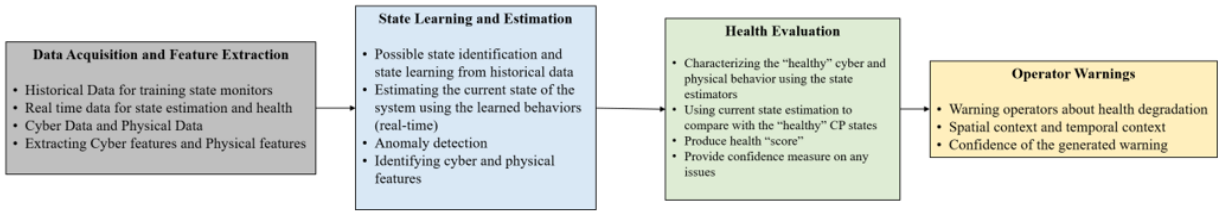

Fig 1: Health monitoring/measurement framework and its components

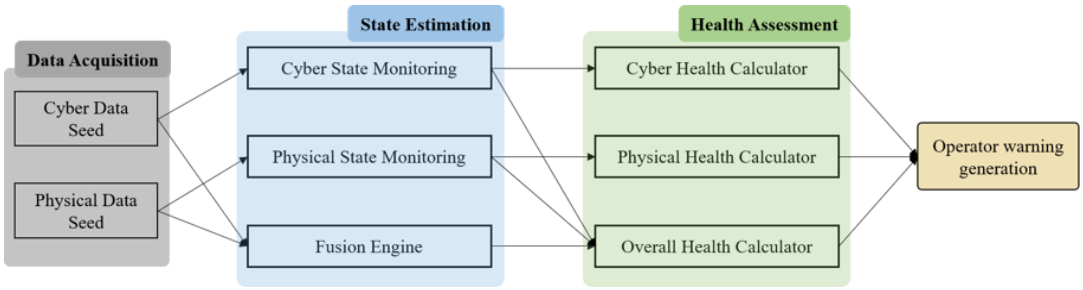

Fig 2: Data flow of the health measurement framework

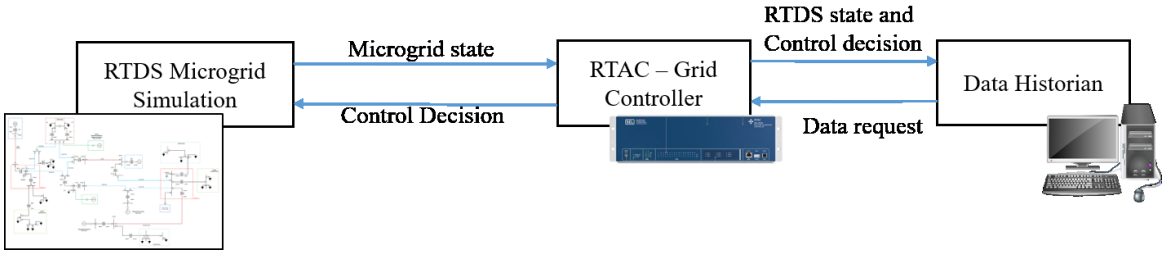

Fig 3: Test bed for implementing the CPS health measurement framework

testbed operate in. Cyber states and Physical states can be identified separately through separate analysis. Further, a fusion engine needs to be developed to take a holistic view at the cyber-physical system.

The state learning algorithms learn the patterns that exist in data and identifies similar behavior. This process is used to characterize the "normal" behavior for cyber and physical aspects using the extracted features in the previous section. An unsupervised approach or a supervised approach could be used to learn the states and their behavior. The advantage of using unsupervised approaches is that the states could be identified without any prior knowledge about the data. The similarities and dissimilarities in data can be used to group the data into different states.

Mainly, in this context, the need is to identify "normal"/"healthy" behavior and any state which shows a degradation of health. In a cyber-communication context, the degradation could be a cyber-intrusion. In a physical power system context, that can be a physical attack or any other sub-optimal behavior such as a component failure. When all the states are correctly identified using the historical data, these state information can be used to estimate the state of the cyber-physical system in real time using real time data streams.

C. Cyber-Physical Health Evaluation

The goal of this phase is to produce a single figure that indicates the operational health of the CPS. The objective is to measure the physical health, cyber health and an overall health of the system. This calculation leverages the learnt states in the state learning phase.

In the state learning phase, the different states that the system goes through can be identified. In that, in a controlled environment such as the testbed, data collected for the "normal" operations can be used as the baseline to characterize the "healthy" state(s) of the system in both cyber and physical sense. In this paper, since the focus is on cyber communications, the DNP3 communication data that are collected during normal operations is used to identify the "healthy" cyber baseline. Deviation from the identified baseline is considered as a degradation of health. This calculation is done in real-time.

D. Warning Generation

Once the health is calculated in real time, if there is a degradation in health, warnings is generated for the operator in human understandable linguistic terms. The warnings should have temporal and spatial context, i.e. when and where the degradation is happening.

III. Algorithms Used in Implementation

This section provides a brief background on the machine learning algorithms used in the current implementation of the presented framework. First, Self-Organizing Maps (SOMs) are presented, then One-Class Support Vector Machines (OCSVM) are discussed.

A. Self-Organizing Maps

The Self-Organizing Map (SOM) was developed by Kohonen [9], [10]. SOMs employ unsupervised learning and are comprised of a topological neuron grid usually arranged in a 2D grid. The main function of the SOM is that its capability of mapping high dimensional input spaces into a low dimensional space while maintaining the topological relationships. The SOM employs competitive learning to learn relationships in data. SOMs have been successfully used in many areas including image clustering and classification, speech recognition , process control, telecommunication and robotics[11], [12].

The learning of the SOM is carried out using the following steps:

Step 1 – Initialization: Assuming that the SOM is a 2D grid of neurons of size $n \times m$, each neuron of the grid maintains a weight vector of dimensionality *d,* where *d* is the dimensionality of the input space. Each weight vector is initialized randomly.

Step 2 – Sampling: An input pattern is selected randomly from the training data.

Step 3 – Competitive learning: The best matching unit (BMU) for the selected input pattern is selected. The BMU is selected by calculating the Euclidean distance to all the neurons from the input patterns and taking the neuron with the minimum distance. The BMU for a input pattern b_g can be expressed as,

$$BMU(b_g) = \underset{j}{\operatorname{argmin}} \left\| b_g - w_j \right\| \quad (1)$$

where, w_j is the weight vector of the jth neuron.

Step 4 – Cooperative weight update: In this step, the weights of the BMU and its neighbors are updated. The idea is to move the neurons that are similar, close together. The weight update of the jth neuron is expressed as follows:

$$w_j(i+1) = w_j(i) + \alpha(i)\, h_j(i)(b_g - w_j(i)) \quad (2)$$

where *i* is the iteration, α, h are the learning rate and the degree of membership to a neighborhood centered at the BMU.

Step 5 – Convergence test: In this step, the convergence criterion is checked. If the criterion is met, the training is stopped. If not, algorithm starts from **Step 2**

B. One Class Classification

One-class learning is to learn the specifics of a single class of data [13]. In this problem, one-class learning is beneficial to perform learning of the "normal"/"healthy" behavior. A One-Class S

Therefore, one-class classification problems are ideal for anomaly detection problems where the determination is whether the data record belongs to the "normal" behavior or not. Support Vector machines are classifiers based on statistical learning techniques [14]. They have been successfully used in many research areas including face detection and recognition, information retrieval, image retrieval, handwritten character recognition, prediction and natural language processing [14]. In [15], researchers have suggested a methodology for adapting the SVM classifiers to one-class classification problems. In this work, the one-class classification problem is tackled using a one-class support vector machine (OCSVM).

IV. IMPLEMENTATION

This section presents the current implementation of the presented health framework components. The current implementation involves implementation of the first three components of the framework.

A. Implementation Data Acquisition and Feature Extraction

In the current implementation of the presented framework. Only the cyber aspect is considered. Cyber data were collected from the testbed without any external disturbances to characterize normal behavior and during cyber intrusions that were created. The collected network data were processed to extract features. The extracted features were used to train the learning algorithms. In the initial version of the implementation, unsupervised learning algorithms were used for data exploration and state identification.

Feature extraction was carried out on the DNP3 packet stream using a windowing technique. The DNP3 packet stream is considered as a time series and a set of statistical features is extracted from a set of neighboring packets by using a window of length one second. Therefore, the set of neighboring packets generate one window based feature vector. The feature extraction is shown in Fig 4. 9 Features were extracted from each window: 1) speed of communication (packets per second), 2) Speed of data (bytes per second), 3) average time gap between the packets in the window, 4) Average number of packets with the same destination address, 5) Number of source addresses, 6) average window size for the packets, 7) number of packets with zero window size, 8) average length of data per packet and 9) maximum data length per packet. This results in a dataset with data records consisting of 9 features that can be used to characterize the different behavior of the cyber communications and can be used to detect intrusions and other health degradations.

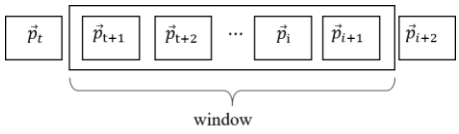

Fig 4: Window based feature extraction

B. Implementation of State Identification and State Estimation

In this implementation, initial data exploration and state learning is carried out using Self-Organizing Maps (SOMs) and One-Class Support Vector Machines (OCSVM). These algorithms are used mainly because of their unsupervised learning capabilities. SOMs have the capability of providing a low dimensional embedding of high dimensional data. Therefore, the 9-dimensional space can be embedded in a two-dimensional space while preserving the topological characteristics in data. Therefore, SOMs enable visualizations of high dimensional data. Further, SOMs are a proven clustering algorithm that can group data based on their similarities. SOMs based visual data mining techniques can be used to extract relationships that exist in features [12], [17].

C. Implementation of Cyber-Physical Health Evaluation

Using the identified states in state learning phase, the health of the cyber-physical system can be measured. Once the healthy or normal behavior is characterized using the features that was extracted, the deviation from that can be considered as the health degradation.

For instance, a vector that is representative of the "healthy" behavior ($\overrightarrow{v_h}$) can be calculated. This vector can be a cluster center, or a simple average of the features for the data records which are in the "healthy" state. Once $\overrightarrow{v_h}$ is obtained using historical data, the real-time cyber-physical health can be calculated using the deviation from $\overrightarrow{v_h}$.

In this implementation, the deviation is calculated using the Euclidean distance. For every time period *t*, the feature vector ($\overrightarrow{v_t}$) is generated in real time using the procedure explained Section IV-A, then, the Euclidean distance to $\overrightarrow{v_h}$.is calculated as follows:

$$d_t = \|\overrightarrow{v_h} - \overrightarrow{v_t}\|_2 \quad (3)$$

The degree of membership to the "healthy" state of the system is considered as the real time health of the system in this implementation. To calculate this degree of membership, a Gaussian neighborhood is used to quantify the health of the system. The distance d_t is used to define a Gaussian neighborhood with mean $\overrightarrow{v_h}$.and variance σ^2:

In this implementation, $\overrightarrow{v_h}$ is set by calculating the average of the data that are identified to be "normal", i.e. the data points that are within the boundary defined by the OCSVM.

V. EXPERIMENTS

This section presents the experiments conducted to test the feature extraction and the state

In this paper, SOMs are used to learn the states that the system goes through during operations. As mentioned, only cyber data are considered in this study. In order to capture the possible cyber states, packet streams were collected for normal communications. Further, cyber data were collected during a Denial of Service (DoS) attack which was introduced on the test bed.

Fig 5 shows the behavior of each dimension extracted from the trained SOM. It can be clearly seen that the *communication speed* and *the average packets targeted at the same destination* are highly correlated (circled in the image). This information can be used to infer the state of the data in the clusters with those attributes. For instance, a window with higher speed and a higher number of communication connections to the same destination can be an indication of a DoS attack

Fig 6 shows the U-Matrix view of the trained SOM. The U-Matrix is a 2D visualization of how far the neurons are to each other in the output space. Therefore, the U-Matrix can be used to identify the topological behavior of the dataset. Each neuron is a representation of several data points where the neuron was chosen as their BMU. Therefore, neurons that are closer together in the U-Matrix indicates that the data records are closer together as well. In the U-Matrix view presented, it can be seen that there are four different states that the system operates in. Bulk of the data fall within the red lines shown on the U-Matrix. Therefore, that can be assumed as the most prevalent state and the "normal" state in the cyber communication. It can be observed that few

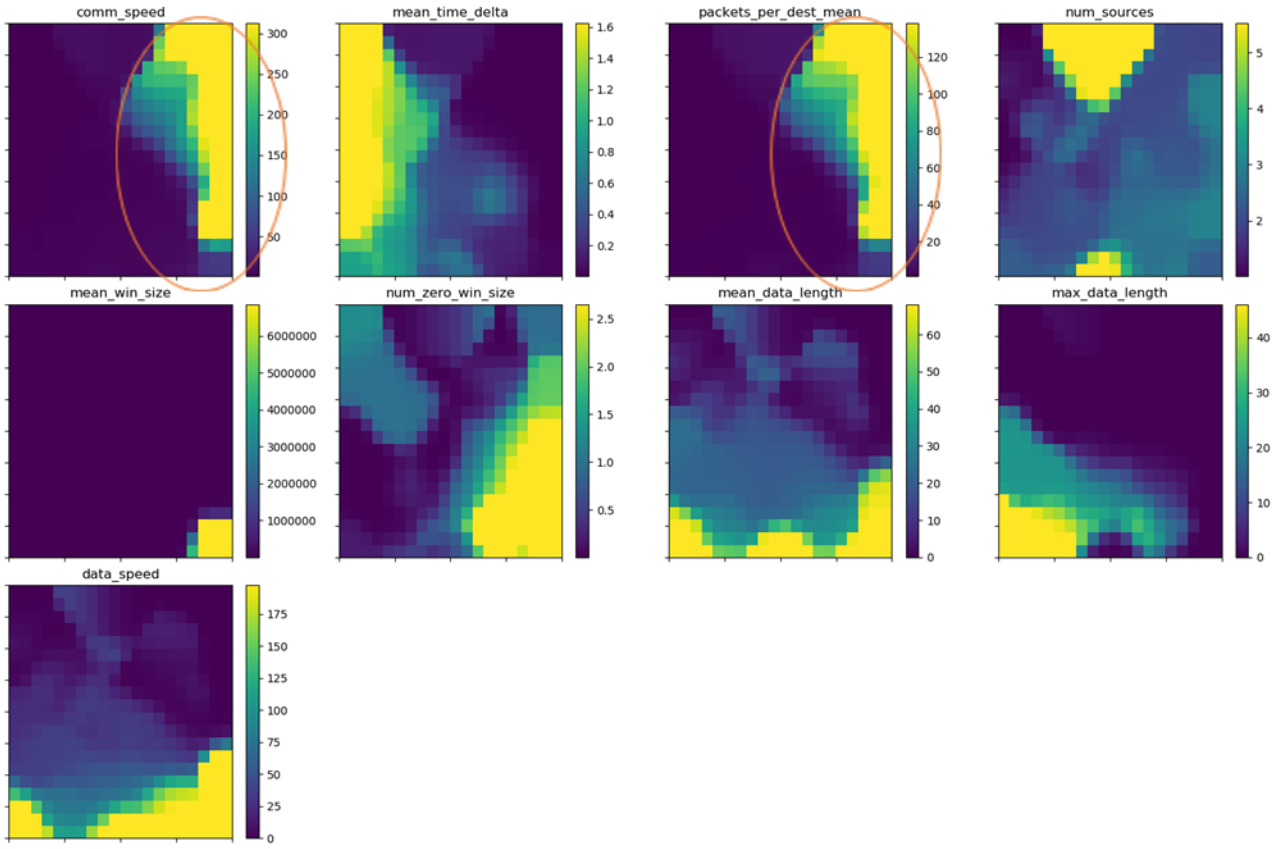

Fig 5: Feature behavior in the output space as shown by the SOM. It can be seen that the communication speed and the packets per destination are highly correlated

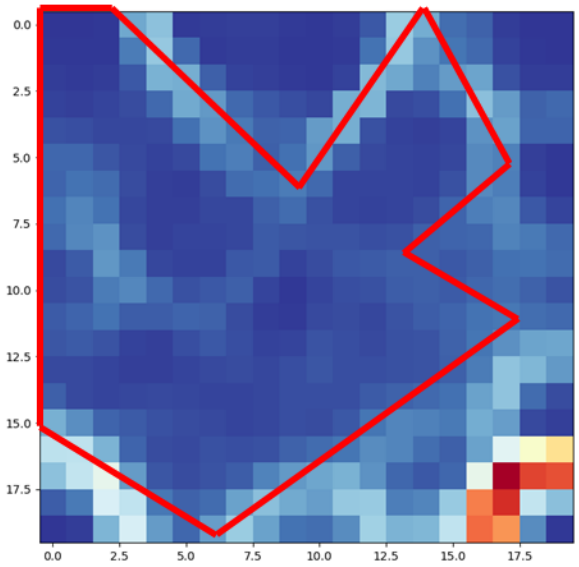

Fig 6: The U-Matrix view of the trained SOM. The darker places indicate areas close together. Lighter places indicate areas apart. The red shows areas which are well separated. The redlines show the possible cluster separations or different states

data points are far away from the rest (bottom right corner). Those can be attributed as the extreme anomalies. Further, when comparing the U-Matrix with the behavior of input dimensions, it can be seen that there is a strong correlation between feature values and the data patterns that lie outside of the enclosed area in red. Further analysis is needed to verify the classes of the data points that fall under these clusters.

Fig 7 and 8 show the results of the health calculation for normal communication and a communication record which contained a introduced DoS attack on the test bed. It was noticed that the health for the normal communication oscillated between 0 and 100 continuously. However, during the attack, there was a clear degradation of cyber health. The health calculation methodology should be modified to remove these oscillations. Further, the health degradation has to be gradual, so that the operators can take measures for mitigation.

VI. Conclusions and Future Work

This paper presented a framework for monitoring the health of Cyber-Physical Systems. The framework consists of four main components 1) data acquisition and feature extraction, 2) state identification and real time state estimation, 3) cyber-physical health calculation and 4) operator warning generation. Further, this work presented an initial implementation of the first two components. The implementation was carried out on a testbed consisting of a Microgrid simulation, a controller, data historian and a cyber-network connecting the physical components. Data acquisition and state identification was carried out for cyber data. A windowing feature extraction technique was used and a SOM was used to perform the state learning. The SOM based visualizations indicated that the there were about four different states and some features were highly

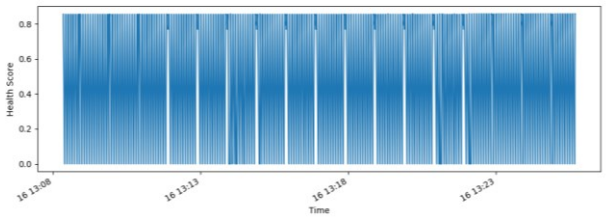

Fig 7: Cyber health calculation of a previously known normal communication. The health oscillates between 0 and 100.

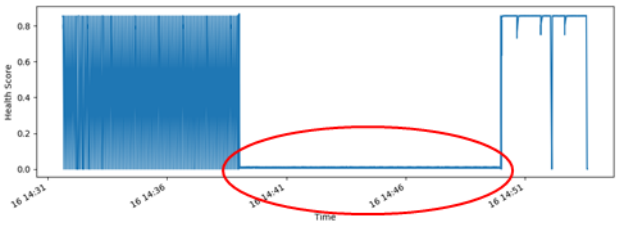

Fig 8: Cyber health calculation of a communication which contained a DoS Attack. The health degradation is circled in red

correlated with the clusters shown by the SOM. Further studies need to be carried out to analyze the clusters produced by the SOM. As next steps, the framework will be refined using the testbed and the data analytics will be extended to physical data as well.

REFERENCES

[1] J. Shi, J. Wan, H. Yan, and H. Suo, "A survey of Cyber-Physical Systems," *2011 Int. Conf. Wirel. Commun. Signal Process. WCSP 2011*, 2011.

[2] Y. Zhang, I.-L. Yen, F. B. Bastani, A. T.Tai, and S. Chau, "Optimal Adaptive System Health Monitoring and Diagnosis for Resource Constrained Cyber-Physical Systems," *2009 20th Int. Symp. Softw. Reliab. Eng.*, pp. 51–60, 2009.

[3] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting Mobility for Energy Efficient Data Collection in Wireless Sensor Networks," *Mob. Netw. Appl.*, vol. 11, no. 3, pp. 327–339, Jun. 2006.

[4] G. Wu, J. Sun, and J. Chen, "A survey on the security of cyber-physical systems," *Control Theory Technol.*, vol. 14, no. 1, pp. 2–10, 2016.

[5] R. (Raj) Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," 2010, p. 731.

[6] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[7] G. Hackmann, Weijun Guo, Guirong Yan, Zhuoxiong Sun, Chenyang Lu, and S. Dyke, "Cyber-Physical Codesign of Distributed Structural Health Monitoring with Wireless Sensor Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 63–72, Jan. 2014.

[8] J. Lee, B. Bagheri, and H.-A. Kao, "Recent Advances and Trends of Cyber-Physical Systems and Big Data Analytics in Industrial Informatics," p. 6, 2014.

[9] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[10] T. Kohonen, "The self-organizing map," *Neurocomputing*, vol. 21, no. 1–3, pp. 1–6, Nov. 1998.

[11] M. Khanum, T. Mahboob, W. Imtiaz, H. A. Ghafoor, and R. Sehar, "A Survey on Unsupervised Machine Learning Algorithms for Automation, Classification and Maintenance," *Int. J. Comput. Appl.*, vol. 119, no. 13, pp. 34–39, Jun. 2015.

[12] C. S. Wickramasinghe, K. Amarasinghe, and M. Manic, "Parallalizable deep self-organizing maps for image classification," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–7.

[13] S. S. Khan and M. G. Madden, "A Survey of Recent Trends in One Class Classification," in *Artificial Intelligence and Cognitive Science*, Springer, Berlin, Heidelberg, 2009, pp. 188–197.

[14] H. Byun and S.-W. Lee, "Applications of Support Vector Machines for Pattern Recognition: A Survey," in *Pattern Recognition with Support Vector Machines*, Springer, Berlin, Heidelberg, 2002, pp. 213–236.

[15] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the Support of a High-Dimensional Distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

[16] "Wireshark Go Deep." [Online]. Available: https://www.wireshark.org/. [Accessed: 06-Jun-2018].

[17] D. Wijayasekara, O. Linda, and M. Manic, "CAVE-SOM: Immersive visual data mining using 3D Self-Organizing Maps," in *The 2011 International Joint Conference on Neural Networks*, 2011, pp. 2471–2478.