

## Online Modules to Introduce Students to Solar Array Control using Neural Nets

**Vivek Sivaraman Narayanaswamy, Arizona State University**

Vivek Sivaraman Narayanaswamy received his B.E degree in electronics and communication engineering at S.S.N College of Engineering, Anna University, Tamil Nadu, India, in 2017. He is currently a graduate student in the school of electrical, computer and energy engineering at ASU, Tempe, AZ. He completed an internship with Qualcomm R&D this summer. His current research interests include developing automated techniques for topology reconfiguration in PV panels to maximize power and to investigate other possible metrics and deep learning frameworks that can be used for speaker diarization.

**Mr. Uday Shankar Shanthamallu**

"I received my B.S degree in Electronics and Communications from the National Institute of Engineering, India in 2011. I am currently pursuing my Master's and PhD program in Electrical Engineering at Arizona State University(ASU). I am advised by Dr. Andreas Spanias. I joined Sensor, Signal and Information Processing Center (SenSIP) at ASU in Jan 2016. My research interests lie at the overlap of sensors and Machine learning and Big Data including, but not limited to Pattern recognition and Anomaly detection. In summer 2016, I did a summer internship at NXP Semiconductors where I worked on sensor data analytics for anomaly detection. I worked on integrating machine learning algorithms on an embedded sensor systems for Internet of Things applications, which can identify anomalies in real time. Before joining ASU, I worked as Systems engineer for 4 years at Hewlett Packard Research and Development, Bangalore, India."

**Abhinav Dixit, ARIZONA STATE UNIVERSITY**

I am a PhD student at School of Electrical, Computer and Energy Engineering at Arizona State University. My research interest includes early detection of neurological diseases through irregularities in speech. I also work as a Research Assistant at SenSIP Center, ECEE at ASU. I am currently involved in developed of JDSP HTML5, interactive DSP software developed in HTML5.

**Mr. Sunil Rao**

**Prof. Raja Ayyanar, Arizona State University**

Raja Ayyanar received the M.S. degree from the Indian Institute of Science, Bangalore, India, and the Ph.D. degree from the University of Minnesota, Minneapolis. He is presently an Associate Professor at the Arizona State University, Tempe. His current research activities are in the area of power electronics for renewable energy integration, dc-dc converters, power management, fully modular power system architecture and new control and pulse—width modulation techniques. He received an ONR Young Investigator Award in 2005.

**Prof. Cihan Tepedelenlioglu, Arizona State University**

**Prof. Andreas S Spanias, Arizona State University**

Andreas Spanias is a professor in the School of Electrical, Computer, and Energy Engineering at Arizona State University. He is also the founder and director of the SenSIP industry consortium. His research interests are in the areas of adaptive signal processing, speech processing, and audio sensing. He and his student team developed the computer simulation software Java-DSP (J-DSP - ISBN 0-9724984-0-0). He is author of two text books: Audio Processing and Coding by Wiley and DSP; An Interactive Approach. He served as associate editor of the IEEE Transactions on Signal Processing and as General Co-chair of IEEE ICASSP-99. He also served as the IEEE Signal Processing vice-president for conferences. Andreas Spanias is co-recipient of the 2002 IEEE Donald G. Fink paper prize award and was elected Fellow of the IEEE in 2003. He served as distinguished lecturer for the IEEE Signal processing society in 2004.

**Dr. Mahesh K Banavar, Clarkson University**

Mahesh K. Banavar is an assistant professor in the Department of Electrical and Computer Engineering at Clarkson University. He received the BE degree in Telecommunications Engineering from Visvesvaraya Technological University, Karnataka, India in 2005, the MS degree and the Ph.D. degree, both in Electrical Engineering from Arizona State University in 2007 and 2010, respectively. His research areas are Signal Processing and Communications, Sensor Networks, and Engineering Education.

**Ms. Sameeksha Katoch**

**Ms. Emma Pedersen**

**Dr. Photini Spanias**

**Pavan Turaga**

**Farib Khondoker, SenSIP Center - Arizona State University**

# **Online Modules to Introduce Students to Solar Array Control using Neural Nets**

## **Abstract**

The growth in the field of machine learning (ML) can be attributed in part to the success of several algorithms such as neural networks as well as the availability of cloud computing resources. Recently, neural networks combined with signal processing analytics have found applications in renewable energy systems. With machine learning tools for solar array systems becoming popular, there is a need to train undergraduate students on these concepts and tools. In our undergraduate signal processing classes, we have developed self-contained modules to train students in this field. We specifically focused on developing modules with built-in software for applying neural nets (NN) to solar array systems where the NNs are used for solar panel fault detection and solar array connection topology optimization which are essentially ML classification tasks. We initially developed software modules in MATLAB and also developed these models on the user-friendly HTML-5 JavaDSP (J-DSP) online simulation environment. J-DSP allows us to create and disseminate web-based laboratory exercises to train undergraduate students from different disciplines, in neural network applications. In this paper, we describe our efforts to enable students understand the properties of the main features of the data used, the types of ML algorithms that can be applied on solar energy systems, and the statistics of the overall results. The modules are injected in our undergraduate DSP class. The project outcomes are assessed using pre and post quizzes and student interviews.

## **Introduction**

The introduction of machine learning algorithms to optimize renewable energy generation is emerging as an important topic in university education and industrial research. Although there are distinct courses in solar energy generation and machine learning, the creation of content at the overlap of these two areas offers several opportunities for education and research. Machine learning [1] promises to solve several problems in solar energy generation including a) fault detection [2,3,4,5], b) shading prediction [6], and c) topology optimization [7,8,9]. Preparing students early in their plans of study to tackle these problems requires: a) training in machine learning, b) exposition to solar energy systems simulation [10], c) skill building in terms of developing or using software to integrate machine learning to obtain solar system analytics and control the overall system [11].

In this paper, we describe an educational program developed to bring to undergraduate classes [12,13] select topics on the utility of machine learning in solar energy generation. The program consists of online modules developed specifically to: a) provide machine learning training to undergraduate students and b) show how ML can be applied to improve the performance of solar energy systems. The program was motivated in part by our Cyber-physical system research [6] and projects developed in our REU site [14,15] in which we embedded undergraduate students in machine learning tasks for solar energy applications. In our program, we used online modules in undergraduate classes and also adapted these modules for use in high school outreach.

We describe in the paper online machine learning software that we developed specifically for online education purposes. This software is based on the HTML 5 J-DSP environment [16] which can run on any browser and is accessible freely by students and faculty. We also describe the content that accompanies the J-DSP software modules and computer exercises. To assess the modules and software, we develop pre- and post- quizzes which we provide to students before and after the module experience.

The rest of the paper is organized as follows. Section 2 describes an overview of the functionalities of the solar monitoring system developed by the SenSIP Center [3] and the need for signal processing and machine learning techniques for solar array analytics. Section 3 introduces machine learning algorithms that have been utilized in the exercises on solar array systems. Section 4 and 5 elucidates the exercise on fault detection and connection topology reconfiguration on solar array panels using machine learning. Section 6 provides the proposed evaluation method of the exercises given to the students and section 7 provides the conclusion and summarizes the contributions of the paper.

### **Description of the Solar Monitoring and Control System**

The SenSIP Center has developed an 18kW solar array cyber-physical system research facility which consists of 104 PV panels [3]. Figure 1a and 1b show the solar array research facility with “smart monitoring devices” (SMDs) on each individual panel for monitoring the PV array. Every panel has an SMD associated with it. Each SMD has dedicated sensors that take measurements from every panel, actuators that can change the connections, and a radio that enables communications with a central hub and eventually the Internet. Sensor data includes voltage, current, irradiance and temperature. Data received from each SMD attached on every panel will be used to obtain analytics [17] and train ML algorithms for fault detection and solar array connection topology reconfiguration. The availability of sensor data from a solar site has great educational value as the students can learn about the behaviour of solar panels relative to environmental and other conditions.

Each SMD has relay switches that enable real time dynamical reconfiguration of the PV array. i.e., an underperforming PV module can be bypassed or reconfigured in certain topologies to maximize power output. Students can learn that reconnecting solar panels in different configurations under shading can improve efficiency.

The SMDs also have Zigbee wireless communicators which allow for wireless transmission of data to an access point or fusion centre. The data from the panels is then transmitted wirelessly over the Internet thus allowing obtaining analytics and enabling remote control of the entire array.

Thus, we explore the possibility of fault detection and identification and array topology reconfiguration remotely. The functional overview of our CPS solar monitoring system is illustrated in Fig.2. It can be seen that by using machine learning algorithms such as neural networks that

leverage data from the panels along with a server, issues such as solar panel fault detection and array topology reconfiguration can be solved with better ease involving fewer manual interactions.



Fig.1: The Cyber physical (CPS) system allowing transmission of analytics and enabling the control of the system from remote sites. The figure shows (a) Solar array research facility with SMDs attached on each panel. (b) The SenSIP solar array facility. The educational value of this CPS facility is that students can learn about sensors and sensor fusion, the use of machine learning to optimize solar energy production.

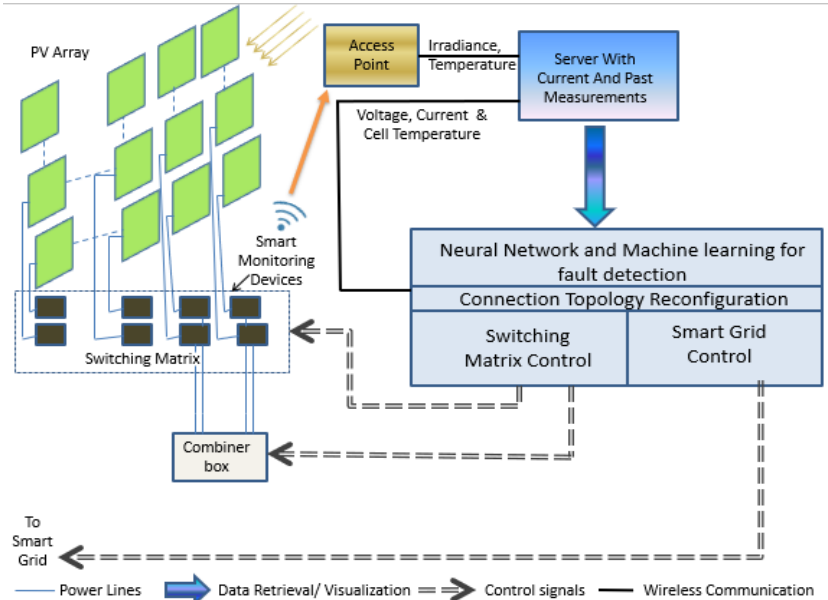


Fig. 2. Overview diagram of the CPS solar monitoring system. The CPS systems consists of sensors on each panel which measure the voltage, current, irradiance and temperature of the module. This data is transmitted wirelessly to the fusion centre allowing for fault detection and topological reconfiguration. Students learn the workings of a true CPS system and the utility of computers and ML in controlling solar energy production.

Recent work [2,6,7,17] shows how various machine learning and signal processing techniques are used to detect faults, predict shading, and select connection topologies. Certain machine learning methods provide a generalized system that learns a useful mapping function between the input features and the output. The algorithms can also “learn” the non-linearities of the data which is specifically useful for solar array analytics due to the non-linear PV characteristics. In the case of solar panel fault detection, algorithms such as neural networks can classify a variety of dependent faults with high accuracy [5]. On the other hand, machine learning algorithms [8] can provide an end-to-end system with reduced number of switching between the panels, as in the case of the connection

topology reconfiguration of solar panels. The advantages of machine learning systems mentioned above emphasize the need for such algorithms for solar analytics. The exposition of undergraduate students to the basics and properties of ML algorithms through the solar energy application offers several opportunities for building skills in using intelligent algorithms for green energy applications. In the following sections, we introduce our modules in machine learning.

### **Module on Machine Learning Algorithms**

There are several applications of ML including Internet of Things (IoT) [1], Energy [24], Smart home [25], natural language [26] and biomedical systems [27]. These applications are becoming part of everyday life and it is important for students to learn about the enabling ML technology.

Learning from data enables a system to make decisions and activate devices to accomplish specific objectives. For example, in energy related applications one can train an ML model to detect faults or inefficiencies and then command the actuators embedded in the SMDs to reconnect modules and optimize the system or make it more resilient. Also, students can learn how ML algorithms often learn and reveal hidden insights from the data that is unknown otherwise.

Many complex engineering tasks run ML algorithms in the background which attempt to understand and model system behaviour. By understanding both system characteristics and ML algorithm properties, one can provide an effective solution for a particular task. With this intent, we introduce various ML algorithms to undergraduate DSP students at an early stage in their curriculum in order to enable them to match algorithms to applications. Algorithms such as k-means clustering and Artificial Neural Networks (ANN) can be introduced and compared, and subsequently applied to fault detection and topology reconfiguration in PV arrays.

#### **'K'-Means Clustering**

K-means clustering aims to partition  $n$  observations into  $k$  clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. Since undergraduate students in the DSP class have already taken courses such as, random signal analysis, exposition to k-means methods is possible. Given a set of observations  $(x_1, x_2, \dots, x_n)$ , where each observation is a  $d$ -dimensional real vector, k-means clustering aims to partition the  $n$  observations into  $k(\leq n)$  sets  $S = \{S_1, S_2, \dots, S_k\}$  so as to minimize the inter-cluster sum of squares (ICSS) (sum of distance functions of each point in the cluster to the  $K$  centre:

The standard K-means equation is given, as follows:

$$\arg \min \sum_{x \in S_i} \sum_{i=1}^k |x - \mu_i|^2 \quad (1)$$

where  $\mu_i$  is the mean of points in set  $S_i$ . We assign each observation to the cluster whose mean yields the least within-cluster sum of squares (WCSS).

$$S_i^{(t)} = \{x_p : \left\| x_p - m_i^{(t)} \right\|^2 \leq \left\| x_p - m_j^{(t)} \right\|^2 \forall j, 1 \leq j \leq k\} \quad (2)$$

Each  $x_p$  is assigned exactly to one  $S^{(t)}$ . The centroid is updated in the next iteration as follows

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j. \quad (3)$$

where  $m_i$  is the updated mean.

Figure 3 illustrates the k-means block used for clustering implemented in JDSP. The block allows the user to provide the number of clusters ‘ $k$ ’ as input for clustering. It also provides the values of the mean square error and the cluster centroids.

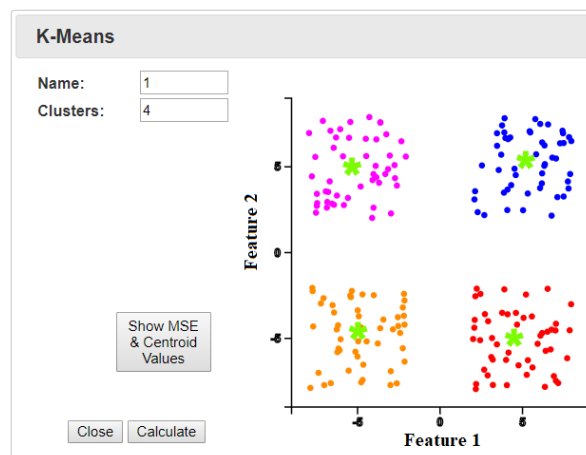


Fig. 3. K-means algorithm implemented in JDSP-HTML5 performed on a custom dataset. The plot shows the centroid for each of the 4 clusters. Students can easily simulate the k-means algorithm both, learning and clustering using J-DSP.

This clustering algorithm can be applied successfully for the detection of certain faults, namely ground faults and arc faults in a solar array [17]. However, as the number and type of faults increase among the PV modules, the k-means clustering algorithm cannot be used to clearly identify the type and nature of faults, facilitating the need for other algorithms such as neural networks.

### Neural Networks

The introduction of neural networks after covering FIR filters in the Undergraduate DSP class is also possible. With the advent of deep learning paradigms, several neural network architectures have been developed [18-23]. Since neural networks are capable of learning various features from the data, hand designed features are not necessarily required. Artificial Neural Networks are comprised of several layers of stacked neurons. A simple neuron takes in weighted inputs from neurons in a previous layer, computes the sum as shown in Fig. 4. The sum is compared with a threshold and outputs a positive value if greater than the threshold. Recently, thresholds are replaced by a family of non-linear functions known as activation functions. The complex classification capabilities of neural

nets can be explained by considering an XOR dataset as shown in Fig.5. The data points in red belong to the negative class and data points in blue belong to the positive class.

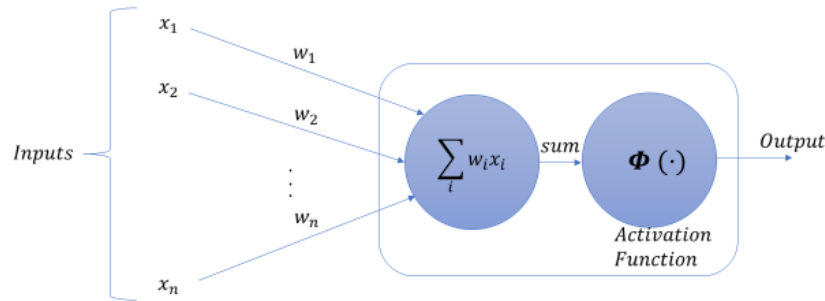


Fig. 4. Neural network thresholding process. The linear combination of the signals before the activation functions can be understood easily by undergraduate DSP students that have already covered FIR filters.

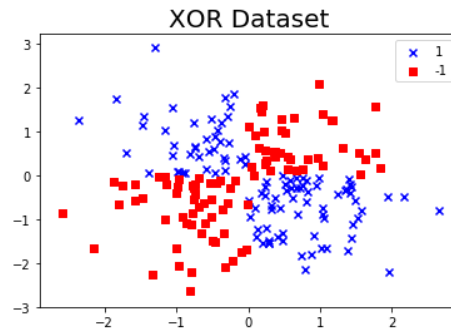


Fig. 5: XOR dataset. Algorithms such as support vector machines or logistic regression are not able to explicitly classify this data. Neural networks learn complex non-linear decision boundaries to classify such data. This paradigm again introduces the utility of thresholds and non-linear functions for learning cluster boundaries.

When a classification algorithm such as logistic regression which provides a linear decision boundary is employed, to classify the XOR data, it is impossible to draw a decision boundary using a straight line on XOR data. Whereas a simple 3-layer ANN can easily classify the data points in the XOR data. This is illustrated in Fig. 6. ANNs can learn a complex non-linear decision boundary because of the non-linear activation functions used in the neurons. And by stacking layers of neurons, the neural net is able to learn multiple levels of abstractions of the data and can classify data accurately.



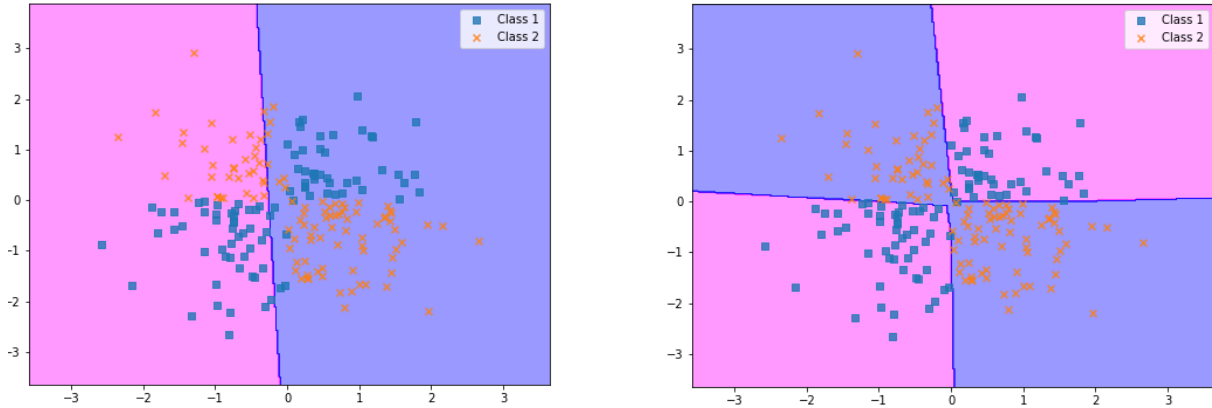


Fig. 6: Comparison between the decision boundaries learned by logistic regression (left) and the ANN (right). Logistic regression is unable to classify the XOR dataset whereas the ANN is able to correctly classify the dataset.

This powerful capability of neural networks can be utilized for solar panel fault detection and connection topology optimization. It can learn and classify the data with a high degree of accuracy for both the applications. The neural network module implemented in J-DSP will enable the students to understand the different properties and parameters of the network and be able to classify faults and topologies given the training and test datasets.

### **Module and Exercise on Fault Detection using Machine Learning**

One of the most basic unsupervised methods for ML is the k-means clustering. It has extensive applications in IoT systems [24], and more recently solar analytics [3]. The introduction of k-means in education, using our award-winning J-DSP was described in [11,16]. In this paper, we discuss how k-means is introduced in our classes in the context of solar energy monitoring and control.

We have used the k-means algorithm as part of our Cyber Physical systems project [6] and have described a method to detect and characterize solar array faults [4,5,17]. In this education project, we form a J-DSP simulation of k-means for fault detection to present to class for the purpose of showing how ML is used in solar energy systems. The results obtained using k-means is shown in Fig. 7.

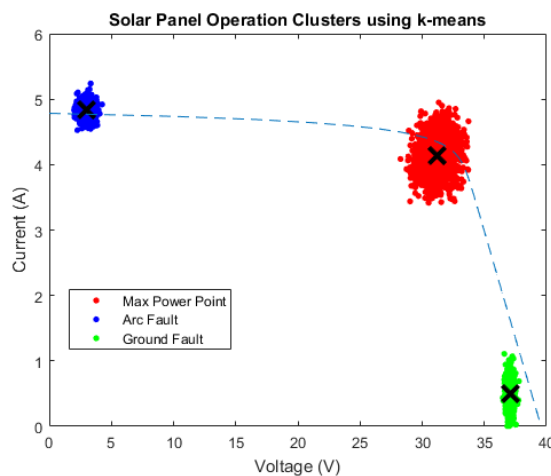


Fig. 7 Fault detection using the k-means algorithm. The algorithm forms clusters of data relative to I-V curves of a solar array. The blue and the green clusters represent the arc and the ground faults while the red cluster represents panels functioning under normal conditions at the maximum power point. When the data clusters away from the maximum power point, the system underperforms. The students here learn about the solar panel I-V curves and how k-means classification can be used to detect an underperforming or faulty panel.

While the k-means clustering algorithm can be used for fault detection, they fail when the variance in the dataset is high [5]. Also, several commonly occurring faults and shading conditions are not identified by k-means algorithm. This is primarily due to the lack of labels in the dataset. We need an algorithm which uses partially labelled data to classify unlabeled data. Using neural networks allows not only detection but identification of the fault type with a high accuracy. The following section discusses the implementation of neural networks for fault detection experiments in J-DSP.

### MLP Model for Solar Array Fault Detection

In this section, we show how an MLP model can be used for solar array monitoring and control [4]. Figure 8 shows a specific form of the Multilayer Perceptron (MLP) neural network model. The overall goal of this model is to selectively distribute normalized weights between 0 and 1 to the given input values for a neuron. Then, a weighted sum propagates through the network, and a projected error is back-propagated and corrected through the network for a certain number of epochs. After various iterations of propagation and backpropagation, the network will provide a final classification of states. The outputs of each “neuron” in a given layer are fed into the input of the next layer. Each neuron has an associated input value  $x_i$  multiplied by a weight  $w_i$ , where  $i$  is the layer index. The weighted sum from each neuron in a layer is added to a bias  $b$  and finally fed into an activation function, which determines the output of the relevant neuron within the next layer.

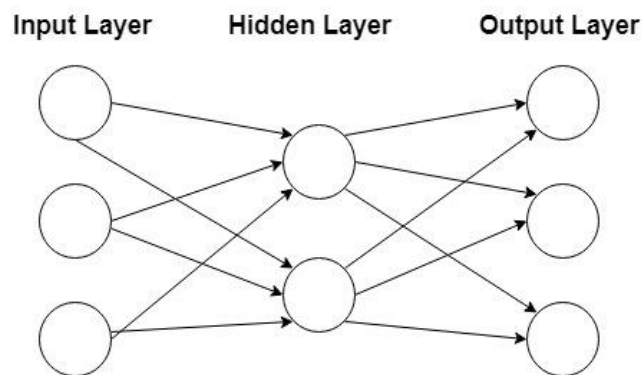


Fig. 8: Simplified, representation of a Multilayer Perceptron Model (MPM). This model will be used for solar array fault detection. Students will use J-DSP to simulate an MPM using a J-DSP module.

The MLP will be trained to detect and classify faults. Types of decisions made include: no fault, an arc fault, or a ground fault. Given that there are sixteen different panels, sixteen unique classifications will be made. Note that due to the series nature of each string, redundancy can occur between faults depending on the direction of the current. In Fig. 9, the fault classifications are color-coded, where green is normally operational, black is a ground fault, and red is an arc fault. This

simulation method is shown for various situations utilizing synthetically created data with added Gaussian noise. This allows us to model real-time solar PV panel data received in the field.

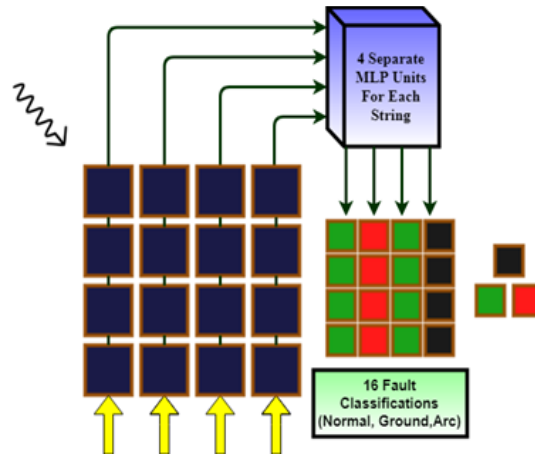


Fig. 9 Overall layout of the PV panel fault detection simulator. The solar panels above are shown in the given 4x4 representation. Each graph represents an I-V curve (horizontal voltage in volts, vertical current in amps). The graphs and vertical strings match up from left to right.

For the simulation process, synthetic data was created with MATLAB. Each panel within every vertical string is represented by 20 I-V points generated as random variables with Gaussian noise. The addition of noise allows us to mimic real I-V conditions for a given PV panel. Thus, a total of 320 points are shown within 16 different clusters, each cluster representing a different panel. Within the simulation, panel clusters are shown together in each graph, but a specific panel's values can be shown via a drop-down menu. Students will use the fault detection module using MLP programmed in JDSP as illustrated in Fig. 10.

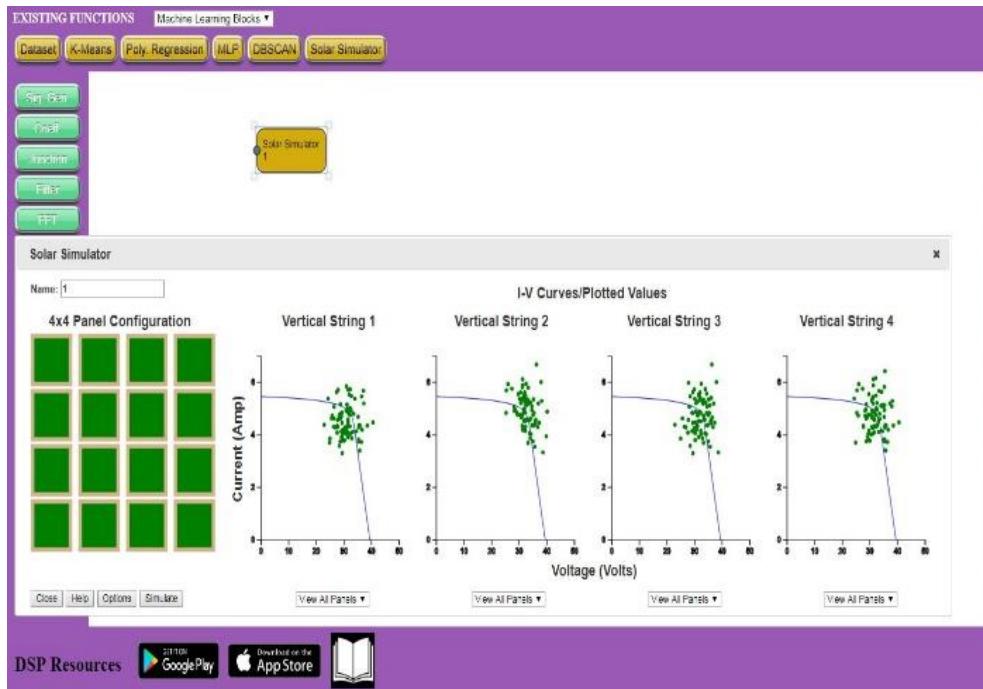


Fig.10: JDSP simulation of fault detection using Neural Networks. The figure shows a 4\*4 array setup developed in JDSP with Maximum Power Points (MPPs) lying at the knee of the IV curve. Faults can be identified using neural networks due to the presence of labels in the dataset. In this simulation, we can identify MPPs from ground faults and arc faults.

### **J-DSP Module and Exercise on Topology Optimization using Neural Nets**

The overall performance and power output of a PV array system depends on factors such as varying temperature, PV panel faults, soiling of the panels, and partial shading conditions. Among these factors considered, the power output reduces significantly due to partial shading of the panels. Modern cyber-physical systems for PV array control and monitoring such as the system installed by the SenSIP Center, consists of ‘Smart Monitoring Devices’ (SMDs) as described in section 2. The SMDs in addition to providing data from the panels can also behave as switches which can electrically reconfigure the existing PV array topology. Power improvement under partial shading conditions can be achieved by modifying the connection topology of the PV panels [7] and the SMDs can be leveraged to perform the same. Popular reconfiguration strategies involve reconfiguring the entire array into Series-Parallel (SP) or Total Cross Tied (TCT) topologies [9]. The SP topology is the conventional connection topology in PV arrays where the panels are connected in series to form strings and strings are connected in parallel to form the array. On the other hand, in a TCT topology, the PV panels are connected in parallel to form rows which are connected in series with the other rows. The two topologies are illustrated in Fig.11a and 11b. Although the cross tied topology has additional redundant connections in between, it has been shown to improve the overall power output under certain scenarios [7, 25].

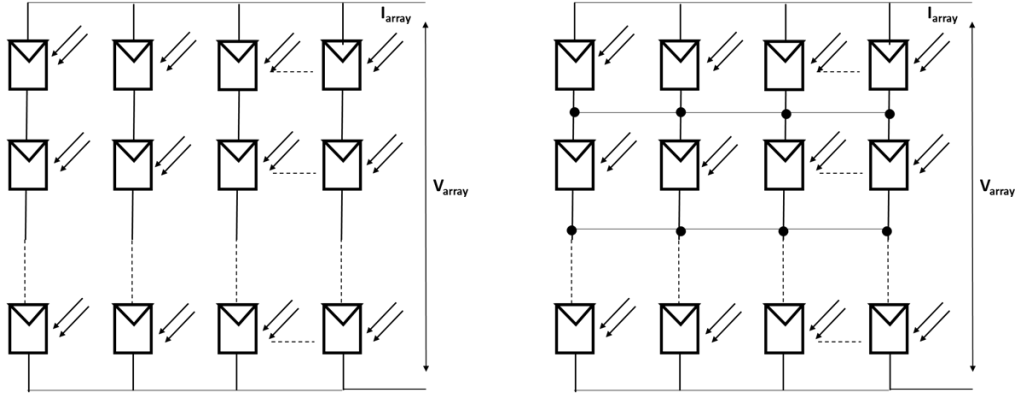


Fig. 11: a) Series-Parallel, b) Total Cross Tied connection topologies.

The concept of connection topology optimization can be viewed from a machine learning perspective. An ML model for such an application can provide an end-to-end, generalizable system which can automatically learn a function to map the input features and outputs. Such models can be easily incorporated in CPS systems providing additional functionalities. In the exercise provided, students perform connection topology reconfiguration using neural networks on a PV array consisting of 12 panels. The input to the neural net is the set of irradiance values measured from each PV panel of the array while the output is the label, which is essentially the topology that improves the power generated under shading conditions. The neural network used is a two-layer feedforward neural network. This exercise is essentially a classification problem and provides an opportunity for students to understand basic concepts and provide a thought process of how neural networks can be used for other applications relating to renewable energy. Figure 12 describes the overall algorithm using a flow chart.

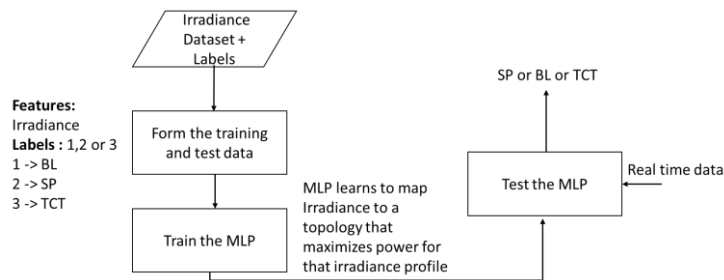


Fig. 12: Flowchart describing the overall algorithm. The diagram illustrates the overall methodology which involves pre-processing, training and classification. Students are taken step-by-step through a J-DSP simulation and observe how the MLP is trained from solar data.

This exercise on neural network classification will be developed on the HTML5- JDSP software which supports a web-based simulation setup. JDSP provides a simple drag and drop type of modules providing simplicity and ease of use. Figure 13 depicts the basic module diagram of the exercise from JDSP. ‘Dataset 1’ block consists of the training dataset which is essentially an  $M \times N$  matrix where  $M$  is the number of observations and  $N$  is the irradiance features of the panels. The block also contains a vector of labels corresponding to every observation. The ‘Dataset 2’ block contains the test dataset

and the corresponding ground truth labels. The ‘Neural Network’ block contains a feedforward neural network that is trained on ‘Dataset 1’ and tested on ‘Dataset 2’. The ‘Accu-Analysis’ block provides the accuracy of classification on the test data set.

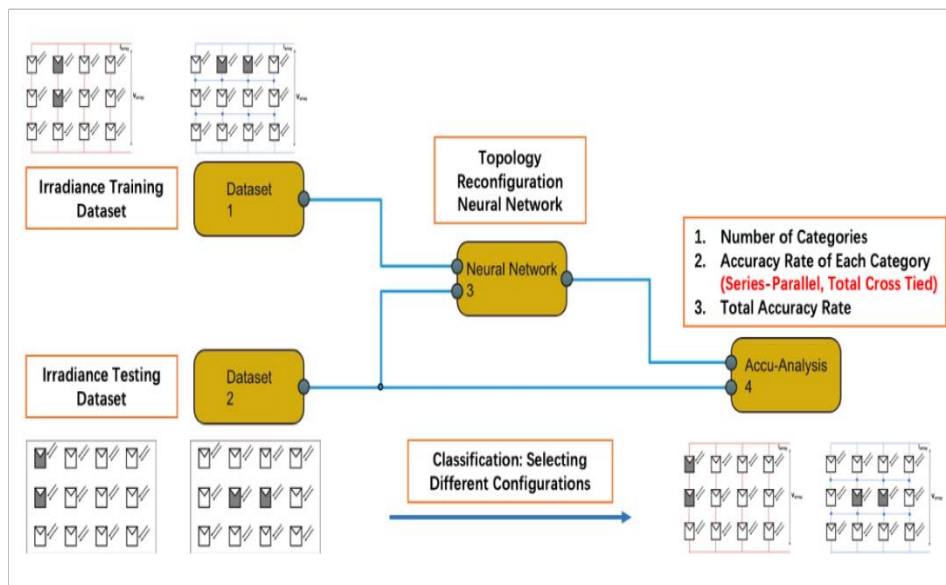


Fig. 13: JDSP simulation of connection topology optimization using Neural Networks. It can be understood that the trained neural network model takes irradiance as input and outputs the recommended topological configuration. An example use case is given in the bottom. Students will use the program for selecting the appropriate topology for different irradiance conditions.

All the exercises and learning modules are used in our DSP class. Portions of the training modules have also been used in our REU site program (Fig. 14) to train undergraduate students in sensors and machine learning. We also plan use of our modules in an upcoming international research experiences program at the University of Cyprus.



Fig. 14. REU students and faculty at the SenSIP solar site. Students have been introduced through a module to machine learning and sensors for solar array monitoring.

## **Assessment and Evaluation**

The main aim of these exercises is to enable students and researchers to obtain a hands-on experience on relevant machine learning applications for solar energy systems namely fault detection and connection topology reconfiguration. The students are initially introduced to the basic concepts of solar panel behavior, types and faults and possible array reconfiguration strategies. They are also given a detailed summary of machine learning algorithms such as k-means and neural networks through two online lectures and are then assigned these JDSP exercises. After working on these tasks, students can relate the concepts learnt to real life applications. A pre-quiz will be given to the students before assigning these exercises as a class project and we refer to this quiz as pre-assignment quiz. The assessment questions are based on the k-means algorithm, basic concepts of neural networks and PV panel behavior. Multiple-choice questions and True or False based questions were asked. A post-quiz will also be given to the same class after completion of the exercise which essentially comprises of the same set of questions given to the students earlier. The results of the pre and post assessment quiz will be presented at the conference.

## **Conclusion**

In this paper, we present the usage of JDSP-HTML5 simulation environment in developing exercises on PV system monitoring and control applications using neural networks for use in the undergraduate DSP class. Using the JDSP-HTML5, one can perform various DSP operations and apply machine learning functions on different datasets. We have illustrated the concepts and importance of fault detection and array reconfiguration from a machine learning perspective using synthetic data. We discussed a variety of other algorithms in machine learning and elucidated K-means clustering algorithm and neural networks in detail. We also discussed the basic blocks to perform the exercise on the JDSP-HTML5 online environment. Finally, we provided some initial assessments.

**Acknowledgement:** This educational research is funded in part by the NSF IUSE program and the NSF CPS Grant #1646542.

## **References**

- [1] U. S. Shanthamallu, A. Spanias, C. Tepedelenlioglu and M. Stanley, "A brief survey of machine learning methods and their sensor and IoT applications," *2017 8th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pp. 1-8, Larnaca, August 2017.
- [2] H. Braun, S. T. Buddha, V. Krishnan, A. Spanias, C. Tepedelenlioglu, T. Takehara, S. Takada, T. Yeider, and M. Banavar, Signal Processing for Solar Array Monitoring, Fault Detection, and Optimization, *Synthesis Lect. Power Electronics, J. Hudgins, Ed. Morgan & Claypool*, vol. 3, Sep. 2012.

- [3] A. Spanias, C. Tepedelenlioglu, E. Kyriakides, D. Ramirez, S. Rao, H. Braun, J. Lee, D. Srinivasan, J. Frye, S. Koizumi, Y. Morimoto, "An 18 kW Solar Array Research Facility for Fault Detection Experiments," *Proc. 18th MELECON*, Tech. Co-sponsor IEEE Region 8, T1.SP1.12, pp. 1-5, Lemassos, April 2016.
- [4] F. Khondoker, S. Rao, A. Spanias and C. Tepedelenlioglu, "Photovoltaic Array Simulation and Fault Prediction via Multilayer Perceptron Models," *2018 9th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pp. 1-5, Zakynthos, Greece, 2018.
- [5] S. Rao, A. Spanias, C. Tepedelenlioglu, "Solar Array Fault Detection using Neural Networks", *2<sup>nd</sup> IEEE International Conference on Industrial Cyber-Physical Systems (ICPS 2019)*, Taipei, May 2019.
- [6] S. Katoch, G. Muniraju, S. Rao, A. Spanias, P. Turaga, C. Tepedelenlioglu, M. Banavar, D. Srinivasan, "Shading prediction, fault detection, and consensus estimation for solar array control," *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pp. 217-222, St. Petersburg, 2018.
- [7] H. Braun, S. T. Buddha, V. Krishnan, C. Tepedelenlioglu, A. Spanias, M. Banavar, and D. Srinivasan, "Topology reconfiguration for optimization of photovoltaic array output," *Elsevier Sustainable Energy, Grids and Networks (SEGAN)*, pp. 58-69, vol. 6, June 2016.
- [8] V. Narayanaswamy, R. Ayyanar, A. Spanias, C. Tepedelenlioglu and D. Srinivasan, "Connection Topology Optimization in Photovoltaic Arrays using Neural Networks", *2<sup>nd</sup> IEEE International Conference on Industrial Cyber-Physical Systems (ICPS 2019)*, Taipei, May 2019.
- [9] S.T.Buddha, "Topology Reconfiguration To Improve The Photovoltaic (PV) Array Performance", *Master's Thesis Arizona State University*, 2011.
- [10] A. Dixit, U. S. Shanthamallu, A. Spanias, V. Berisha and M. Banavar, "Online Machine Learning Experiments in HTML5," *2018 IEEE Frontiers in Education Conference (FIE)*, pp. 1-5, San Jose, CA, USA, 2018.
- [11] A. Dixit, U. Shanthamallu, A. Spanias, S. Rao, S. Katoch, M. Banavar, G. Muniraju, J. Fan, P. Spanias, A. Strom, C. Pattichis, H. Song, "Multidisciplinary Modules on Sensors and Machine Learning," " *Proc. 2018 ASEE Annual Conference*, Salt Lake City, Utah, June 2018.
- [12] A. Spanias, *Digital Signal Processing; An Interactive Approach – 2nd Edition*, 403 pages, Textbook with JAVA exercises, ISBN 978-1- 4675-9892-7, Lulu Press On-demand Publishers Morrisville, NC, May 2014.



- [13] A. Spanias, V. Atti, A. Papandreou-Suppappola, K. Ahmed, M. Zaman, T. Thrasyvoulou, "On-line signal processing using J-DSP", *IEEE Trans. Signal Process. Lett.*, vol. 11, no. 10, pp. 1-5, Oct. 2004.
- [14] A. Spanias, J. Blain Christen, T. Thornton, K. Anderson, M. Goryll, H. Arafa, U. Shanthamallu, E. Forzani, H. Ross, W. Barnard, S. Ozev, "The Sensor Signal and Information Processing REU Site, " *Proc. 2018 ASEE Annual Conference*, Salt Lake City, Utah, June 2018.
- [15] A. Spanias and J. B. Christen, "A Stem Reu Site on the Integrated Design of Sensor Devices and Signal Processing Algorithms," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6991-6995, Calgary, AB, 2018.
- [16] A. Dixit, S. Katoch, P. Spanias, M. Banavar, H. Song and A. Spanias, "Development of signal processing online labs using HTML5 and mobile platforms," *2017 IEEE Frontiers in Education Conference (FIE)*, pp. 1-5, Indianapolis, IN, 2017.
- [17] S. Rao, S. Katoch, P. Turaga, A. Spanias, C. Tepedelenlioglu, R. Ayyanar, H. Braun, J. Lee, U. Shanthamallu, M. Banavar, D. Srinivasan, "A Cyber-Physical System Approach for Photovoltaic Array Monitoring and Control," *Proceedings 8th International Conference on Information, Intelligence, Systems and Applications (IEEE IISA 2017)*, pp.1-6, Larnaca, August 2017.
- [18] E. Rosten., T. Drummond, "Machine Learning for High-Speed Corner Detection." *ECCV*, vol 3951. Springer, Berlin, Heidelberg, 2006.
- [19] Alm, C. Ovesdotter, D. Roth, and R. Sproat. "Emotions from text: machine learning for text-based emotion prediction." In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pp. 579-586. Association for Computational Linguistics, 2005.
- [20] Sebastiani, Fabrizio. "Machine learning in automated text categorization." *ACM computing surveys (CSUR)* 34, pp 1-47, vol. no. 1, 2002.
- [21] Pazzani, Michael J., and Daniel Billsus. "Content-based recommendation systems." In *The adaptive web*, pp. 325-341. Springer, Berlin, Heidelberg, 2007.
- [22] Magoulas, George D., and Andriana Prentza. "Machine learning in medical applications." In *Advanced Course on Artificial Intelligence*, pp. 300-307. Springer, Berlin, Heidelberg, 1999.

- [23] Huang, Zan, Hsinchun Chen, Chia-Jung Hsu, Wun-Hwa Chen, and Soushan Wu. "Credit rating analysis with support vector machines and neural networks: a market comparative study." *Decision support systems* 37, pp. 543-558, vol.no.4, 2004.
- [24] A. Spanias, "Solar Energy Management as an Internet of Things (IoT) Application," *Proceedings 8th International Conference on Information, Intelligence, Systems and Applications (IEEE IISA 2017)*, pp. 1-4, Larnaca, August 2017.
- [25] Picault, D., Raison, B., Bacha, S., Aguilera, J. and De La Casa, J., "Changing photovoltaic array interconnections to reduce mismatch losses: a case study", In *9th IEEE Environment and Electrical Engineering Conference (EEEIC)*, pp. 37-40, May 2010.
- [26] Yang, Rayoung, and Mark W. Newman. "Learning from a learning thermostat: lessons for intelligent systems for the home." *Proc. of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pp. 93-102. ACM, 2013.
- [27] Deng, L., & Liu, Y. (Eds.), *Deep Learning in Natural Language Processing*. Springer, 2018.
- [28] Lee, K. H., & Verma, N., "A low-power processor with configurable embedded machine-learning accelerators for high-order and adaptive analysis of medical-sensor signals," *IEEE Journal of Solid-State Circuits*, 48(7), pp. 1625-1637, 2013.