NILM Dashboard: A Power System Monitor for Electromechanical Equipment Diagnostics

Andre Aboulian, Daisy H. Green, Jennifer F. Switzer, Thomas J. Kane, Gregory V. Bredariol, Peter Lindahl, *Member, IEEE* John S. Donnal, *Member, IEEE* and Steven B. Leeb, *Fellow, IEEE*

***Abstract*—Non-intrusive load monitoring (NILM) uses electrical measurements taken at a centralized point in a network to monitor many loads downstream. This paper introduces NILM Dashboard, a machine intelligence and graphical platform that uses NILM data for real-time electromechanical system diagnostics. The operation of individual loads is disaggregated using signal processing and presented as time-based load activity and statistical indicators. The software allows multiple NILM devices to be networked together to provide information about loads residing on different electrical branches at the same time. A graphical user interface provides analysis tools for energy scorekeeping, detecting fault conditions, and determining operating state. The NILM Dashboard is demonstrated on the power system data from two United States Coast Guard (USCG) Cutters.**

***Index Terms*—Condition-based maintenance, energy efficiency, fault detection, non-intrusive load monitoring**

I. Introduction

Improving energy efficiency of buildings, industrial sites, and military facilities such as Army bases or Coast Guard ships starts with information and insight into the processes involved [1]. Information on the operational patterns and behavior of individual loads often reveals wasteful practices, allowing for effective demand-side energy management vital for emission reduction [2]. Further, faulty mission-critical equipment may operate marginally for extended periods until abruptly failing even though symptoms are often visible in the electrical system weeks before a failure occurs [3]. Thus, monitoring the electrical system for individual load information also allows condition-based diagnostics and prognostics which can minimize the impact of equipment malfunctions or failures in the power system [4].

Non-Intrusive Load Monitoring (NILM) is a rugged, low-cost sensing platform that can fill these needs, providing actionable information in real-time for energy management and equipment diagnostics. NILM sensors are installed at the main power entry or at a subpanel. These sensors measure currents and voltages with either conventional voltage and current transducers or with non-contact sensors that do not require ohmic contact [5], [6]. Measurements are sent to a NILM computer where data is stored in a high-speed time-series database [7] and converted into harmonic power envelopes to facilitate analysis [8]. The NILM computer uses signal processing to detect the operation of individual loads from the aggregate power data and post-processes the electrical measurements for energy score-keeping and diagnostics [9].

Fig. 1. US Coast Guard Cutter ESCANABA [13] and Non-Intrusive Load Monitoring installation onboard the ship

Current NILM research is largely focused on developing increasingly accurate disaggregation techniques using various accuracy metrics [10]. However, gaining user support for smart-metering techniques such as NILM requires more than just accuracy. A successful NILM system must have the ability to provide real-time monitoring and diagnostic analysis, presented in an accessible, easy-to-use format for the end user [11].

The NILM Dashboard introduced in this paper provides a framework for a complete NILM system that captures data, accurately dissagregates load events, analyzes the equipment for potential faults, and presents useful information to end-users in real-time. This dashboard combines novel system architectures, algorithms, and user interfaces to solve not just the technical challenge of accurate load disaggregation, but also several technical challenges more commonly considered in the development of smart meter analytics [11] and industrial internet of things (IIoT) devices and networks [12]. Specifically, the NILM Dashboard simultaneously addresses the challenges of:

- Efficiently collecting, managing, and processing large volumes of raw electrical data
- Accurately disaggregating individual load operation
- Incorporating advanced algorithms for real-time monitoring and diagnostic focused analytics
- Providing users actionable insight into the operation of the electrical system and its individual loads through easy-to-understand visual displays

The NILM Dashboard's utility at addressing these chal-

This work was supported in part by the Office of Naval Research NEPTUNE program, the Grainger Foundation, the MITei-Exxonmobil collaboration, and the Cooperative Agreement between the Masdar Institute of Science and Technology and MIT. A. Aboulian, D. Green, J. Switzer, T. Kane, P. Lindahl, and S. Leeb are with MIT, Cambridge, MA 02139 USA (e-mail: aandre@mit.edu; dhgreen@mit.edu; jfs@mit.edu; tjkane@mit.edu; lindahl@mit.edu; sbleeb@mit.edu). G. Bredariol is with the U.S. Coast Guard, St. Petersburg, FL 33701 USA (e-mail: gregory.v.bredariol@uscg.mil). J. Donnal is with the U.S. Naval Academy, Annapolis, MD 21402 USA (e-mail: donnal@usna.edu).

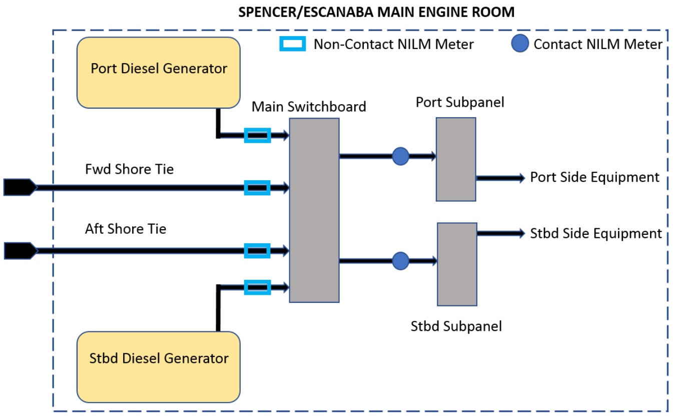


Fig. 2. NILM sensors installed at six locations in the electrical network on a USCG Cutter

lenges is demonstrated through its incorporation into NILM systems installed aboard US Coast Guard Cutters (USCGC) ESCANABA and SPENCER, two 270-ft (82-m) US Coast Guard ships (Fig. 1). On each ship, the NILM system in place consists of four non-contact meters [5], [6] monitoring the overall power consumption of the vessel and two contact meters (featuring conventional current and voltage transducers) monitoring two electrical subpanels (Fig. 2). The contact meters on the subpanels provide high resolution measurements of equipment crucial to the proper operation of ship propulsion, power generation, and auxiliary services. The NILM Dashboard combines data from these meters to act as a Shipboard Automatic Watchstander (SAW), delivering maintenance and fault data to Coast Guard personnel in real-time, ultimately optimizing operations and reducing equipment failures [14], [15]. The system retains favorable aspects of NILM – low sensor count, easy installation, high reliability – but the Dashboard expands capability to provide support for energy scorekeeping, condition-based maintenance, fault detection, and diagnostics. This paper describes the algorithms and structure of the NILM Dashboard and its application on mission critical systems.

II. System Architecture

The NILM Dashboard software stack runs on a typical Linux-based computer. The four layers shown in Fig. 3 are responsible for obtaining measurements, identifying loads, analyzing behaviors, and communicating results. This software stack is designed to address the technical challenges outlined in the introduction, including the ability to work with a large volume of data, accurately disaggregate electrical load events, provide real-time monitoring and diagnostic focused analytics, and the availability of an easy to understand visual display of information [11].

The Dashboard processes information about load *events*, which occur when equipment transitions between ON and OFF states. The event data is classified as a specific load and then mapped to an operating schedule of its activity. This time-based information enables detailed comparison of loads and provides a record of operation. To highlight trends over time, the platform also maintains load *metrics*, which are statistical conclusions that expose anomalies and patterns.

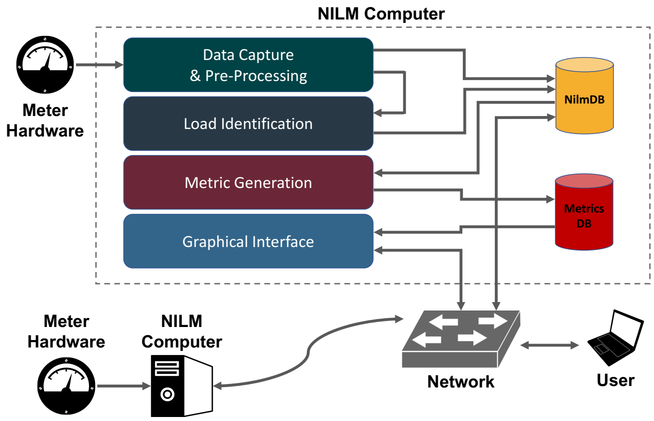


Fig. 3. System architecture of the NILM Dashboard. Rectangular blocks represent software layers, and cylinders represent storage databases. Lines between entities have arrows in the direction of data flow.

A. Data Capture and Pre-Processing

The NILM system captures currents and voltages from metering hardware at a sampling frequency of 3 kHz for non-contact sensors and 8 kHz for contact sensors. These high sampling rates are necessary for capturing transient shapes as loads change state [16] and the higher harmonic content of non-linear loads, e.g. variable frequency drives [17]. Sampling at these rates presents a data volume challenge in terms of processing, storage, and network bandwidth. To reduce this data volume while maintaining the relevant shape and harmonic information, these measurements are pre-processed into harmonic power envelopes using the Sinefit algorithm [8], effectively compressing the high-rate raw current and voltage data into real, reactive, and harmonic power components at a rate congruent with the power system line frequency (60 Hz for the ships). This promotes space-efficiency while maintaining the richness of the original signals. NilmDB is specially suited for storing this time-series data, making it available for high-speed and low-bandwidth access throughout the Dashboard platform [7].

B. Load Identification

The load identification block disaggregates the operation of individual loads from the power data in real-time. The NILM systems installed on the ships feature neural network architectures to achieve highly accurate load disaggregation, described in greater detail in Section IV. The resulting load events encode the type of state transition, as well as the change in real and reactive power levels. NilmDB records these sparse events at the times they occur. The Joule data processing framework [18] is used to streamline the capturing, pre-processing, and identification stages. This robust tool models the data pipeline as a series of processing “modules”, with formally-defined “streams” of information passing between them. Data flows between modules through efficient memory pipes, without needing to access the database as an intermediary. This modular processing framework allows for efficient, real-time monitoring of load events.

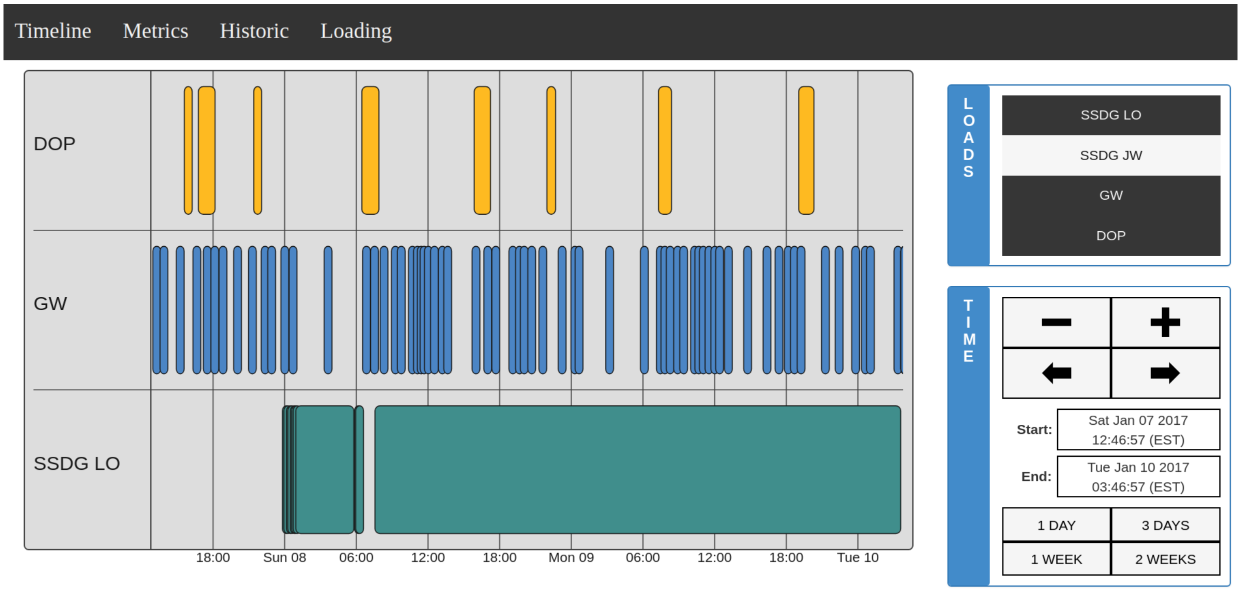


Fig. 4. Timeline interface displaying the status for the Diesel Oil Purifier (DOP), Graywater Pumps (GW) and Ship's Service Diesel Generator Lube Oil Heater (SSDG LO). Colored blocks represent periods when each load was energized.

C. Metrics Generation

To provide actionable insight and diagnostics, the Dashboard must be able to assess the health of equipment in real-time and alert the user to anomalies. This crucial functionality makes the Dashboard an analytic tool, rather than simply a data collection device. To accomplish this, the metric generation block reads event streams from NilmDB and calculates operational metrics correlated to the health of systems and loads [3], [19]. Relevant metrics for a load may include the average power consumption or the number of operations per day. Calculated metrics and their associated metadata, such as type, load, and encompassed time range, are stored in MetricsDB for rapid query from the Dashboard. For the NILM systems on the Coast Guard cutters, six metrics are calculated. However, the MetricsDB must be able to adapt and expand for different NILM systems. The MetricsDB must also have the ability to save large data segments at full resolution, allowing for an anomalous load transient to be stored alongside its respective metrics. To meet these requirements, a NoSQL database structure is employed for MetricsDB [20], [21] due to its easy expandability and flexible size boundaries compared to traditional SQL alternatives.

Metric generation is implemented as a Python-based script that runs automatically every ten minutes. Every time the generator runs, it calculates *rolling* metrics over the past 24-hour window. These values are used to update the gauges in the metrics interface (Section III-B) to reflect current conditions. The metric generator also calculates *daily* load metrics if event data is available in NilmDB for the previous day, which ends at midnight. This information is used to populate the historic view (Section III-C) of the user interface, which shows the daily progression of a chosen metric for a given load.

D. Graphical Interface and Network

The final step is to transform load schedules and metrics into an interactive visual display that is user-friendly, easily-accessible and provides operators with actionable information. This interface is described in detail in the next section. The Dashboard user interacts with a web-based application [22], [23] that can be accessed on the NILM computer or from any computer connected on the same network. The user interface operates exclusively in the client's browser, conserving processing power and network bandwidth. The data visualizations are implemented with a data-driven graphical framework [24].

The browser must retrieve event and metric data to populate the interfaces. NilmDB has external web endpoints, so the client can directly communicate with this database to download event data for a particular load. The MetricsDB, however, cannot be accessed outside of the NILM computer. The web server proxies requests to the MetricsDB by exposing its own web endpoints for the client to reach. Many power systems are sufficiently complicated that more than one NILM system is required to track an entire electrical plant. The Dashboard software provides database access on a local network for NilmDB and MetricsDB, allowing a user to access all the NILM nodes on the local network from one place.

III. User Interface

Providing the information generated by NILM quickly and clearly is paramount to creating a diagnostic tool capable of preventing failures to mission critical equipment. Even after processing, most NILM data is not intuitive to operators unaccustomed to analyzing equipment power streams and transients. The NILM Dashboard addresses this problem with an on-site interface that provides real-time and historic equipment information. To further aid operators in tactical decision making, the Dashboard generates useful metrics on system health. This information is made available to operators through

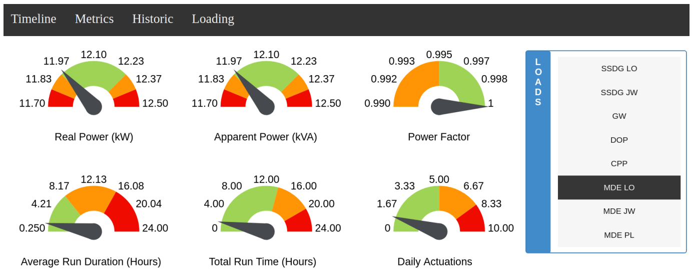


Fig. 5. Gauges indicate whether the metrics pertaining to the selected load are within an acceptable range.

an interface containing four interactive tools: Timeline, Metric View, Historic View, and Loading View.

A. Timeline

The "Timeline", as shown in Fig. 4 provides a live view of the equipment status, allowing the user to see loads activated and secured in real-time. The user can monitor the entire plant or hide certain equipment from view, allowing for increased attention on select loads. The time window can be adjusted to the user's choosing, either through zoom/pan functions or by selecting one of four pre-set time periods. The Timeline tool provides the user with a compact picture of plant operations and the ability to easily investigate any apparent anomalies.

B. Metric View

The "Metric View" shown in Fig. 5 is a user's first stop when a fault is suspected. It provides the user with a set of diagnostic indicators for a selected piece of equipment. The metrics available are real power, apparent power, power factor, average run duration, total daily run time and daily number of actuations. Each metric is displayed as a gauge with green, yellow and red sections. The colored sections are derived from equipment nameplate data, known usage patterns and statistics from previous normal operation. Green indicates normal operations, while yellow and red indicate increasing likelihood of a fault. The gauge needle position is the average metric value for the last 24 hours and is refreshed every ten minutes. The Metric View provides an analysis of individual equipment health and helps direct initial troubleshooting efforts.

C. Historic View

The "Historic View" shown in Fig. 6 provides short and long-term trend data to supplement the analysis from the Metric View. This tool allows users to select a single load and any one of the six metrics listed above. The Historic View is presented as a bar graph that gives the user the ability to track equipment behavior over a period of up to 6 months. While the Metric View is intended for a watchstander to quickly detect a possible fault, the Historic View is designed for the plant manager to assess trend data, track behavior and make decisions on condition-based maintenance.

D. Loading View

The "Loading View" shown in Fig. 7 allows a user to detect phase imbalances and loading discrepancies within the electrical system. The user can select a monitoring point and view the per phase power and total electrical load. In this case study, the Loading View allows the user to compare the total loading of the two generators or the two monitored subpanels. This information can be used for energy-scorekeeping and to optimize power generation.

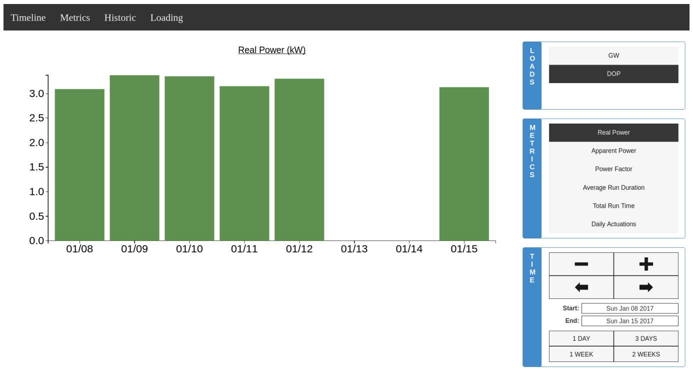


Fig. 6. The Historic View shows the daily trend for a given metric over a period of time.

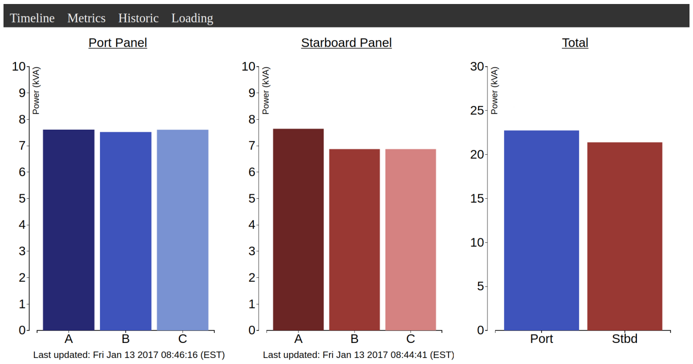


Fig. 7. The Loading View displays the current per-phase levels for a source of power data. With the shown configuration, the port and starboard subpanel are being compared.

IV. Load Identification

Load identification is a key step in the four-stage pipeline. Load identification can be accomplished by many different algorithms, such as artificial neural networks (NN), k-nearest neighbors (k-NN), and correlation-based algorithms [7], [25]. This is where the core of NILM research is focused today. For this application, a neural network approach is taken. Since the two monitored subpanels have a fixed number of loads, a supervised learning approach is used in which data is hand-labeled in order to perform training. As described by Hart [26], there are three main categories of appliances that may be monitored by a NILM system: ON/OFF, Finite State Machine (FSM), and continuously variable. On the studied subpanels there are ON/OFF loads and one FSM load, the Diesel Oil Purifier. An ON/OFF load has only two states, ON or OFF, while a FSM load has several states due to its complex operation. Table I lists all the monitored loads which are to be displayed on the Dashboard.

Load identification occurs in the following three stages:

- **Event Detection** determines when transients occur in the power stream.

- **Event Classification** matches identity of transient to a load.
- **Load Confirmation** checks constraints between load events.

A. Event Detection

After pre-processing, Real (P) and reactive (Q) power are outputted at every line cycle (60 Hz) for each phase. From the fundamental P and Q streams, apparent power (S) is calculated as,

$$S = \sqrt{P^2 + Q^2} \tag{1}$$

The S stream is then smoothed using a 101-point median filter, which eliminates small fluctuations while preserving edges. The S stream is used to detect turn-on and turn-off times by detecting where the stream abruptly changes in value. Converting to apparent power simplifies load detection to a single data stream. The filtered data is convolved against the Laplacian of a Gaussian [27] kernel to compute the smoothed second-derivative of this stream. This effectively maps step changes in apparent power to zero-crossings for easier detection. An empirically-determined threshold is set to remove zero-crossings that are due to small fluctuations of the resulting convolution. A zero-crossing detector is then used to find the location of the steps. Fig. 8 shows several ON and OFF events as detected by the edge detector.

After an ON or OFF event is detected using the S stream, the edge detector examines the P and Q to calculate a set of features to be used as an input vector to the NN. For each phase (A,B,C), represented here as x, an ON event produces four features, the transient-peak ($P_{on,x,peak}$, $Q_{on,x,peak}$) and the steady-state level changes ($P_{on,x,ss}$, $Q_{on,x,ss}$). Because there is no transient peak when a load turns off, an OFF event has only two features per phase ($P_{off,x,ss}$, $Q_{off,x,ss}$). The steady state level change is calculated by taking the difference in the median power level for a defined window (Δt_{window}) before and after an ON/OFF event. The Δt_{window} is chosen to be 0.5 sec, which is 30 samples given a line frequency of 60 Hz. Fig. 9 illustrates these features for the real power stream on one phase. Features for reactive power are calculated using the same process. Equations (2)-(6) generate the features and the input vectors.

TABLE I
MONITORED LOADS IN ENGINE ROOM

Load	Power Rating	Delta Phases	Power Factor	Port Panel	Stbd Panel
Main diesel engine (MDE) keep-warm system					
Lube oil heater (MDE LO)	12 kW	3ϕ	1.0	x	x
Jacket water heater (MDE JW)	9.0 kW	3ϕ	1.0	x	x
Prelube Pump (MDE PL)	2.2 kW	3ϕ	0.82	x	x
Ship service diesel generator (SSDG) keep-warm system					
Jacket water heater (SSDG JW)	7.5 kW	3ϕ	1.0	x	x
Lube oil heater (SSDG LO)	1.3 kW	1ϕ	1.0	x	x
Additional engine room loads					
Controllable pitch propeller hydraulic pump (CPP)	7.5 kW	3ϕ	0.82	x	x
Graywater pumps (GW)	3.7 kW	3ϕ	0.85	x	
Diesel oil purifier (DOP)	5.6 kW	3ϕ	0.85	x	
Oily Water separator (OWS)	6.7 kW	3ϕ	0.90		x

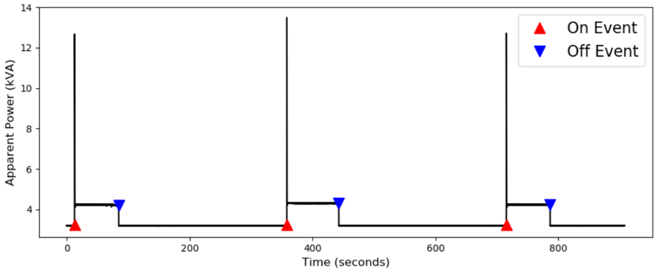


Fig. 8. ON-events and OFF-events for graywater pump runs.

ON event steady state change in power:

$$\begin{aligned} P_{on,x,ss} = &\, median(P_x[t_{end}], ..., P_x[t_{end} + \Delta t_{window}]) \\ &- median(P_x[t_{on} - \Delta t_{window}], ..., P_x[t_{on}]) \end{aligned} \tag{2}$$

ON event transient peak:

$$\begin{aligned} P_{on,x,peak} = &\, max(P_x[t_{on}], ..., P_x[t_{end}]) \\ &- median(P_x[t_{end}], ..., P_x[t_{end} + \Delta t_{window}]) \end{aligned} \tag{3}$$

OFF event steady state change in power:

$$\begin{aligned} P_{off,x,ss} = &\, median(P_x[t_{off}], ..., P_x[t_{off} + \Delta t_{window}]) \\ &- median(P_x[t_{off} - \Delta t_{window}], ..., P_x[t_{off}]) \end{aligned} \tag{4}$$

Here, t_{on} is the time the load turns on, t_{end} is the end of the start-up transient, t_{off} is the time the load turns off, and t_{window} is the length of the window for taking steady state calculations.

The feature input vector for ON events is:

$$\begin{aligned} (&P_{on,A,ss}\ Q_{on,A,ss}\ P_{on,A,peak}\ Q_{on,A,peak} \\ &P_{on,B,ss}\ Q_{on,B,ss}\ P_{on,B,peak}\ Q_{on,B,peak} \\ &P_{on,C,ss}\ Q_{on,C,ss}\ Q_{on,C,peak}\ Q_{on,C,peak}) \end{aligned} \tag{5}$$

The feature input vector for OFF events is:

$$\begin{aligned} (&P_{off,A,ss}\ Q_{off,A,ss} \\ &P_{off,B,ss}\ Q_{off,B,ss} \\ &P_{off,C,ss}\ Q_{off,C,ss}) \end{aligned} \tag{6}$$

There are a fixed number of loads on the panels and the normal real and reactive power draw of each load is known. Therefore the total change in steady state should not be less than the smallest load on the panels. False event detections are reduced by comparing the magnitude of the calculated change in steady state values to a threshold. Events with power value changes beneath this threshold are discarded. In one month (September 2017), there were a total of 2946 events on the USCGC Spencer port panel. The described edge detector correctly found 2936 of those events and incorrectly detected 32 non-events.

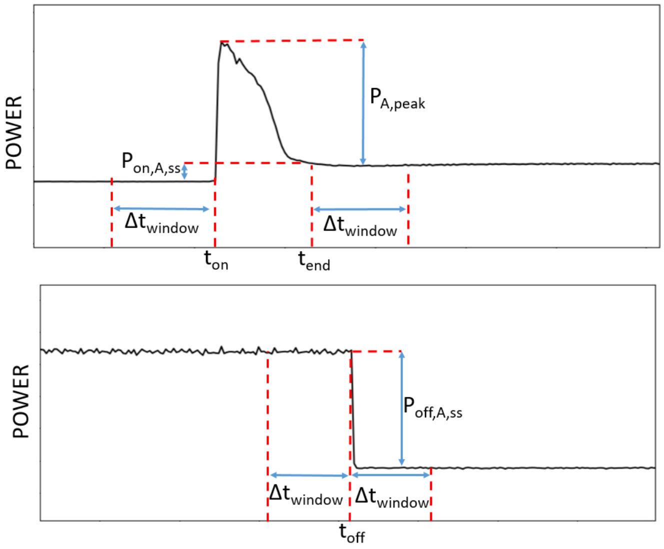


Fig. 9. ON/OFF load features for a single phase

B. Event Classification

To classify each event as an individual load, a fully-connected neural network (NN) using stochastic gradient descent (SGD) is applied [28]. Separate NNs are utilized for ON events and OFF events for each panel. The input layer to the NNs are the features for each load as previously described in (5) for ON events and (6) for OFF events. Two hidden layers of 10 nodes each are used, for which a weighted sum of the inputs from the previous layer to each node is passed through a ReLu (Rectified Linear Unit) [29] activation function, $f(z) = max(0, z)$, where z is the input to a node. Back-propagation is used to find the set of weights and biases to minimize loss. To allow multi-class classification, the output layer is a softmax layer, [30], defined as,

$$f(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e_j^z} \text{ for } i = 1 \ldots k \tag{7}$$

where z is the input to each node, and k is the number of classes. Each ON/OFF load is a unique class. Additionally, some loads often actuate together during normal operations, as they are part of a combined system, creating a new class representing multiple loads. Finally, the Diesel Oil Purifier has a class for each state of operation.The output is a vector of probabilities that sums to one and the classification is made by selection of the class that has the highest probability.

The data was split into three sets: training, validation, and testing. The prediction error of the validation set was used as a stopping criterion during training [31]. To prevent over-fitting, a third separate data set, the testing set was used. Repeated random sub-sampling was performed on the data for 10 iterations, so that for each iteration the data was randomly split by load class into training, validation, and testing data and the performance was evaluated on the testing data.

Some loads have short runs, in the range of minutes, and thus have many events. Other loads have run durations in the range of hours or even days, and thus have far fewer events. To prevent the NN from simply predicting the most common class, the training data had to be better balanced. This was accomplished by random under-sampling of the majority classes on the training and validation data.

TABLE II
ACCURACY OF CLASSIFYING ON-EVENTS

Load	**TP** ± σ_{TP}	**precision** ± σ_P	**recall** ± σ_R
Main diesel engine (MDE) keep-warm system			
LO Heater	8 ± 0	0.92 ± 0.07	1 ± 0
JW Heater	1 ± 0	0.95 ± 0.16	1 ± 0
Prelube Pump	4 ± 0	0.80 ± 0.19	1 ± 0
Ship service diesel generator (SSDG) keep-warm system			
JW Heater	65 ± 0	0.97 ± 0.02	1 ± 0
LO Heater	95 ± 0	0.98 ± 0.01	0.99 ± 0.01
Additional engine room loads			
CPP Pump	4.2 ± 1.5	0.88 ± 0.14	0.84 ± 0.15
Graywater Pump	268.3 ± 1.06	0.99 ± 0	1 ± 0
DO Purifier (Centrifugal)	26 ± 1.05	0.98 ± 0.04	1 ± 0
DO Purifier (Feed Pump)	7 ± 0	1 ± 0	1 ± 0
DO Purifier (Flushing Sequence)	13.9 ± 0.32	0.95 ± 0.04	1 ± 0

It is not sufficient to focus only on the total percentage of correctly classified loads when verifying the accuracy of the identifier. If a load only turns on and off a few times over a month, incorrectly identifying it will not have much effect on the total classification accuracy, but it is still vital that the load be correctly identified. Thus, the accuracy of the model for each class is evaluated by considering the following parameters [32]:

- True Positive (TP): a load event occurs and is correctly identified
- False Positive (FP): a load event is classified, but that event did not occur
- False Negative (FN): a load event occurs but that event is not classified

These parameters are used to determine the classifier's recall and precision, which answer two fundamental questions:
(1) What is the likelihood that a load event is reported? ($recall$)
(2) What is the likelihood that a reported event is correct? ($precision$)

$$recall = \frac{TP}{TP + FN} \tag{8}$$

$$precision = \frac{TP}{TP + FP} \tag{9}$$

Table II shows the results for each load from the 10 iterations. The results presented are the average number of true positives, the precision, and the recall for each load, as well as the standard deviations of each ($\sigma_{TP}, \sigma_P,$ and σ_R respectively). Precision and recall values of one with a standard deviation of zero indicate perfect performance in identifying a specific class. For true positives, a small standard deviation shows that performance between iterations is consistent and that the model is not overfitting.

C. Confirmation and Implementation

Load identification is implemented into the NILM Dashboard architecture through two modules, classification and confirmation. The classification module runs the previously

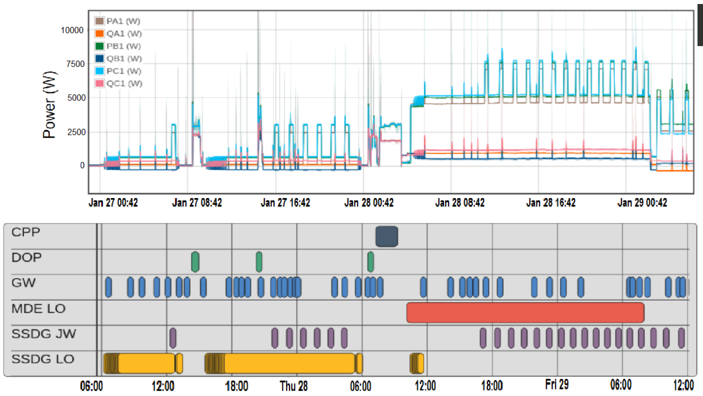


Fig. 10. Load events are shown over a 2-day interval. All the transients present in the origin signals (above) have been detected and are shown in the timeline (below).

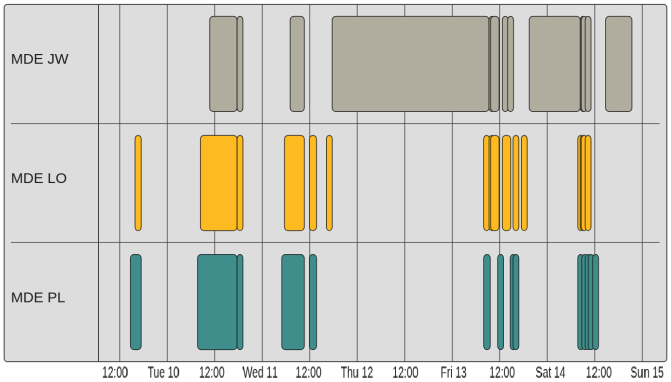


Fig. 11. The MDE system loads displayed with the Timeline over a 5-day period

described detection and classification algorithm on 10 second windowed power stream data. Since there may be events that occur at the edges of an interval, the intervals are overlapped to ensure that no events are lost. For each iteration, the classification module output is piped to a confirmation module. For each load, the confirmation module stores the most recent state (ON or OFF) outputted to NilmDB, to ensure two ON events or two OFF events are not output consecutively. If two consecutive ONs are detected, the confirmation module removes the first occurrence from the NilmDB. If two consecutive OFFs are detected, the second occurrence is not outputted to the NilmDB. This reduces the possibility of the Dashboard incorrectly displaying that a load is energized.

V. Experimental Results

The identification techniques implemented for the NILM Dashboard permit two new applications of nonintrusive power monitoring. First, the experimental results demonstrate the application of these algorithms to provide automatic "watch-standing." The operation of equipment on mission critical systems can be automatically logged to produce activity records normally collected manually by operators. Second, the algorithms presented above are applied in NILM Dashboard to provide immediate real-time indication of system faults based on irregular patterns of power consumption. To validate each layer of the NILM Dashboard architecture, the system was tested end-to-end on busy intervals of underway data from the two US Coast Guard Cutters. This test demonstrates the Timeline's ability to turn a complicated stream into a clear schedule of load events. A power signal and its corresponding Timeline are shown in Fig. 10. The Timeline provides the operator the state of all monitored loads and their recent behavior.

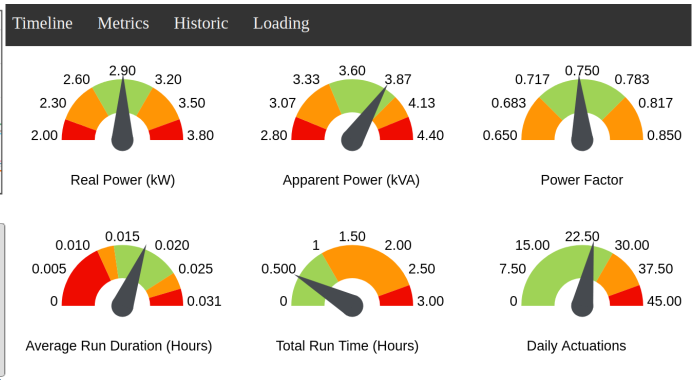


Fig. 12. Healthy graywater pump behavior captured on the Dashboard.

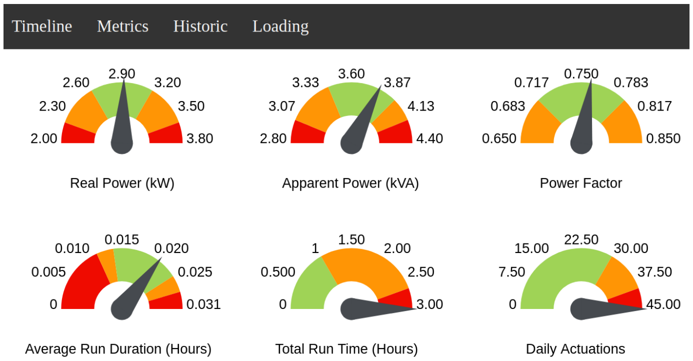


Fig. 13. Dashboard Metrics View displays a fault condition

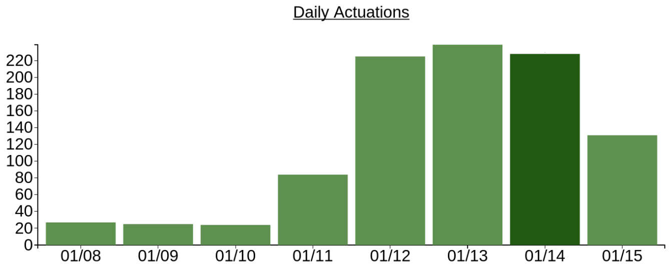


Fig. 14. Historic View of graywater fault condition

The Timeline can be used to monitor the status of interdependent loads, such as the equipment comprising the main diesel engine (MDE) system (jacketwater heater, lubeoil heater, and prelube pump). When the engine is secured, the prelube pump and the lube oil heater should by cycling ON and OFF together. Repeated activation of one without the other may indicate a fault. A Timeline for five days of normal MDE operation while the engine is secured is shown in Fig. 11. This Timeline can be used to produce operational records normally collected by hand, as discussed further below.

The next case-study uses eight days of data from the

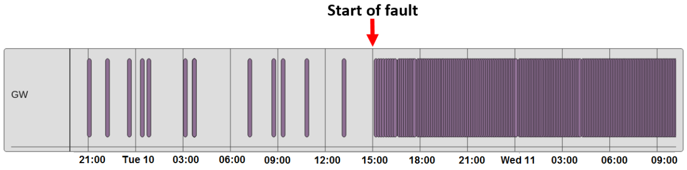


Fig. 15. Timeline view of graywater fault condition

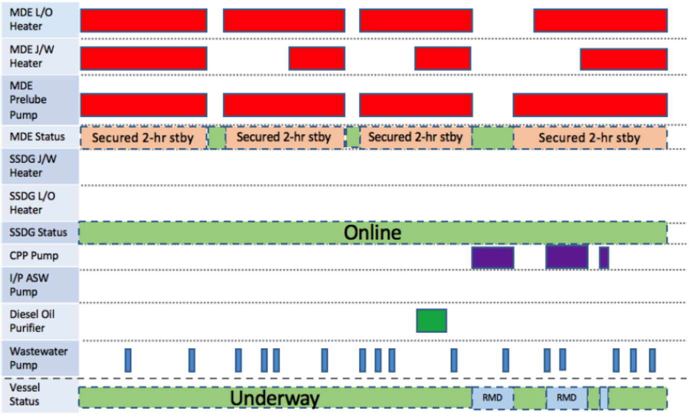


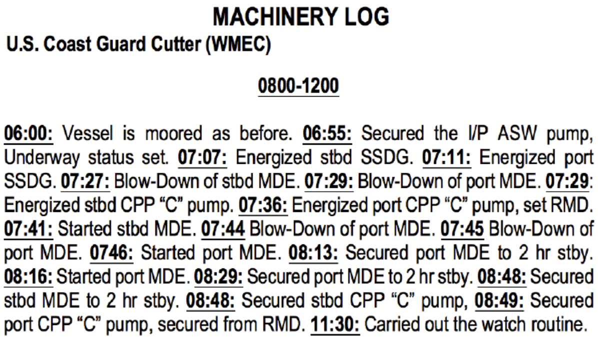
Fig. 16. Load events used as FSM elements for the ship's engines and ship's operational status (below).

SPENCER port-panel. During the first four days of this interval, all equipment behavior was normal. However, during the last four days the graywater pumps entered a fault condition, running ten times more often than normal. The root cause of this fault was later determined to be a broken checkvalve. During normal operation, graywater drains from throughout the ship to the holding tank and the monitored pumps transfer this to a larger storage tank. The broken checkvalve allowed water to flow backwards from the storage tank to the holding tank. The pumps had to operate almost continuously to keep the holding tank from overflowing, a condition that can quickly result in pump failure and the loss of a crucial auxiliary system.

The Dashboard provided clear diagnostic indicators of this fault. Figs. 12 and 13 show the Metrics View from a day of normal graywater pump operation and a day when the pump was backflowing, respectively. For the day of normal behavior, all of the dials are in the green region. On the day that the pump failure occurred, the dials for total daily run time and number of daily actuations jump into the red region due to operation in excess of the pre-established system norms. Fig. 14 displays the total daily run time in the Historic View for the eight days before and during this issue. This view shows a clear trend that the total run time increased significantly after the first four days. The Timeline in Fig. 15 displays precisely when this fault began. With the assistance of a diagnostic tool like the NILM Dashboard, the crew could have detected and responded to the casualty in hours, not weeks.

VI. Observations and Conclusion

The NILM Dashboard provides real-time diagnostics of electromechanical equipment without the need for an extensive sensor network. The platform was successfully tested on power stream data from two Coast Guard cutters, accurately recording machinery behavior and identifying error states. The system is poised to be installed aboard the vessels to assist operators in fault detection, behavior tracking and energy scorekeeping.



MACHINERY LOG

U.S. Coast Guard Cutter (WMEC)

0800-1200

06:00: Vessel is moored as before. **06:55:** Secured the I/P ASW pump, Underway status set. **07:07:** Energized stbd SSDG. **07:11:** Energized port SSDG. **07:27:** Blow-Down of stbd MDE. **07:29:** Blow-Down of port MDE. **07:29**: Energized stbd CPP "C" pump. **07:36:** Energized port CPP "C" pump, set RMD. **07:41:** Started stbd MDE. **07:44** Blow-Down of port MDE. **07:45** Blow-Down of port MDE. **0746:** Started port MDE. **08:13:** Secured port MDE to 2 hr stby. **08:16:** Started port MDE. **08:29:** Secured port MDE to 2 hr stby. **08:48:** Secured stbd MDE to 2 hr stby. **08:48:** Secured stbd CPP "C" pump, **08:49:** Secured port CPP "C" pump, secured from RMD. **11:30:** Carried out the watch routine.

Fig. 17. Automatically generated ship log from NILM data

For greater classification accuracy, there is further research being done on integrating this NN method with other classification methods such as an exemplar shape-matching algorithm [7] and multi-scale median filtering. This would all be done within the classification module, allowing the classification algorithm to be updated without affecting the operation of the user interface.

The next iteration of the NILM Dashboard will use the known loads to directly infer the status of the propulsion plant, or even the whole ship. For instance, the jacket water heaters, lube oil heaters and pre-lube pumps can be used to determine whether the main diesel engine is online, secured or in standby. The main diesel engine essentially becomes a finite state machine, with the stages of operations determined by NILM monitored equipment. With the right indicators, even the operational status of the entire ship can be determined from NILM. For example, the controllable pitch propeller (CPP) pumps are energized when the ship enters a higher state of readiness known as the restricted maneuvering doctrine (RMD). The Timeline View can then display when the ship enters RMD. Fig. 16 shows a Dashboard view with a manually-labeled timeline depicting engine and ship underway status. With insider knowledge of ship operations, NILM data can be used to generate ship's logs as shown in Fig. 17, further reducing the workload of watchstanders.

The Dashboard platform can be adapted to any NILM system, whether it be in a house, factory, or naval vessel, to provide feedback on equipment behavior and energy usage. The NILM Dashboard provides the framework and analysis tools to turn power stream data into actionable information for optimizing operations.

Acknowledgment

The authors gratefully acknowledge the U.S. Coast Guard and the crews of the ESCANABA and SPENCER for granting access to their ships. This work was supported by the Office of Naval Research NEPTUNE program, The Grainger Foundation, the MITei-Exxonmobil collaboration, and by the Cooperative Agreement between the Masdar Institute of Science and Technology (MASDAR Institute), Abu Dhabi, UAE, and the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA - Reference 02/MI/MIT/CP/11/07633/GEN/G/00

References

[1] P. Palensky, S. Member, D. Dietrich, and S. Member, "Demand side management: Demand response, intelligent energy systems, and smart loads," *IEEE Transactions on Industrial Informatics*, pp. 1551–3203, 2011.

[2] F. D. Kanellos, G. J. Tsekouras, and N. D. Hatziargyriou, "Optimal demand-side management and power generation scheduling in an all-electric ship," *IEEE Transactions on Sustainable Energy*, vol. 5, no. 4, pp. 1166–1175, 2014.

[3] J. C. Nation, A. Aboulian, D. Green, P. Lindahl, J. Donnal, S. B. Leeb, G. Bredariol, and K. Stevens, "Nonintrusive monitoring for shipboard fault detection," in *Sensors Applications Symposium (SAS), 2017 IEEE*. IEEE, 2017, pp. 1–5.

[4] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart grid technologies: Communication technologies and standards," *IEEE Transactions on Industrial Informatics*, vol. 7, pp. 529 – 539, 12 2011.

[5] J. S. Donnal and S. B. Leeb, "Noncontact power meter," *IEEE Sensors Journal*, vol. 15, no. 2, pp. 1161–1169, 2015.

[6] J. S. Donnal, P. Lindahl, D. Lawrence, R. Zachar, and S. Leeb, "Untangling non-contact power monitoring puzzles," *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3542–3550, June 2017.

[7] J. Paris, J. S. Donnal, and S. B. Leeb, "NilmDB: The non-intrusive load monitor database," *IEEE Transactions on Smart Grid*, vol. 5, no. 5, pp. 2459–2467, 2014.

[8] J. Paris, J. S. Donnal, Z. Remscrim, S. B. Leeb, and S. R. Shaw, "The sinefit spectral envelope preprocessor," *IEEE Sensors Journal*, vol. 14, no. 12, pp. 4385–4394, 2014.

[9] J. S. Donnal, J. Paris, and S. B. Leeb, "Energy applications for an energy box," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 787–795, 2016.

[10] S. K. Barker, S. Kalra, D. Irwin, and P. D. Shenoy, "Nilm redux: The case for emphasizing applications over accuracy," 2014.

[11] D. Alahakoon and X. Yu, "Smart electricity meter data intelligence for future energy systems: A survey," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 425–436, Feb 2016.

[12] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[13] "United States Coast Guard," https://www.uscg.mil/, accessed: Jan 2018.

[14] G. Bredariol, J. Donnal, W. Cotta, and S. Leeb, "Automatic watchstander through nilm monitoring," in *ASNE Day 2016*. American Society of Naval Engineers, 2016.

[15] G. Bredariol, D. Green, A. Aboulian, J. C. Nation, P. Lindahl, and S. B. Leeb, "NILM: A smarter tactical decision aid," in *Technology, Systems & Ships Day 2017*. American Society of Naval Engineers, 2017.

[16] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong, "Power signature analysis," *IEEE Power and Energy Magazine*, vol. 1, no. 2, pp. 56–63, Mar 2003.

[17] K. D. Lee, S. B. Leeb, L. K. Norford, P. R. Armstrong, J. Holloway, and S. R. Shaw, "Estimation of variable-speed-drive power consumption from harmonic content," *IEEE Transactions on Energy Conversion*, vol. 20, no. 3, pp. 566–574, Sept 2005.

[18] J. Donnal, "Joule: A real time framework for decentralized sensor networks," *IEEE Internet of Things Journal*, 2018.

[19] J. Paris, J. S. Donnal, R. Cox, and S. Leeb, "Hunting cyclic energy wasters," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2777–2786, 2014.

[20] "MongoDB," www.mongodb.com, accessed: Jan 2018.

[21] IBM, "Why nosql? your database options in the new non-relational world," Tech. Rep., Mar 2015. [Online]. Available: https://cloudant.com/wp-content/uploads/Why_NoSQL_IBM_Cloudant.pdf

[22] "Node.js," https://nodejs.org/, accessed: Jan 2018.

[23] "Express – node.js web application web framework," https://expressjs.com/, accessed: Jan 2018.

[24] M. Dewar, *Getting Started with D3: Creating Data-Driven Documents*. OReilly, 2012.

[25] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," in *Sensors*, 2012.

[26] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[27] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.

[28] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 05 2015. [Online]. Available: http://dx.doi.org/10.1038/nature14539

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[30] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*. The MIT Press, 2012.

[31] L. Prechelt, "Early stopping - but when?" in *Neural Networks: Tricks of the Trade*. Springer-Verlag, 1997, pp. 55–69.

[32] P. Klein, J. Merckle, D. Benyoucef, and T. Bier, "Test bench and quality measures for non-intrusive load monitoring algorithms," *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, 2013.

Andre Aboulian graduated from the Massachusetts Institute of Technology in 2018 with a M.Eng. in Electrical Engineering and Computer Science. His interests include embedded systems, distributed software systems, power electronics, computer architecture, and human-computer interaction.

Daisy H. Green received the B.S. degree from the University of Hawaii at Manoa in 2015. She received a M.S. in electrical engineering in 2018 from the Massachusetts Institute of Technology, where she is currently pursuing a Ph.D.

Jennifer F. Switzer received a B.S. degree from the Massachusetts Institute of Technology in 2018. She is currently pursuing an M.Eng. in Electrical Engineering and Computer Science from MIT.

Thomas J. Kane is a Lieutenant in the US Coast Guard currently pursuing an M.S. in Mechanical Engineering at the Massachusetts Institute of Technology. He was previously stationed as Damage Control Assistant aboard USCGC MELLON and as a Port Engineer for the National Security Cutter (NSC) fleet.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2843770, IEEE Transactions on Industrial Informatics

Peter A. Lindahl graduated with his Ph.D. in Engineering from Montana State University in 2013. He is currently a postdoctoral associate in the Research Laboratory of Electronics at the Massachusetts Institute of Technology. His research interests include sensors and instrumentation for energy and power systems, renewable energy generation, and energy policy.

Gregory V. Bredariol is a Lieutenant in the US Coast Guard currently serving as the Engineer Officer aboard USCGC Dilegence (WMEC). He graduated in 2017 from the Massachusetts Institute of Technology with dual M.S. in Mechanical Engineering and Naval Architecture and Marine Engineering.

John S. Donnal received the B.S. degree from Princeton University, Princeton, in 2007, and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, in 2013 and 2016, respectively, all in electrical engineering. His research interests include nonintrusive load monitoring synthesis, energy harvesting, and communications systems. He is currently a Faculty Member of the U.S. Naval Academy in Weapons and Systems Engineering.

Steven B. Leeb received his doctoral degree from the Massachusetts Institute of Technology in 1993. He has been a member on the MIT faculty in the Department of Electrical Engineering and Computer Science since 1993. He also holds a joint appointment in MIT's Department of Mechanical Engineering. He is concerned with the development of signal processing algorithms for energy and realtime control applications.