SimML Framework: Monte Carlo Simulation of Statistical Machine Learning Algorithms for IoT Prognostic Applications

Full/Regular Research Paper, CSCI-ISOT

Ashwini R. More
Oracle Physical Sciences Research Center
Oracle Corporation
San Diego, CA 92121, USA
ash.more@yahoo.co.in

Kenny C. Gross
Oracle Physical Sciences Research Center
Oracle Corporation
San Diego, CA 92121, USA
Kenny.gross@oracle.com

***Abstract*—Advanced statistical machine learning (ML) algorithms are being developed, trained, tuned, optimized, and validated for real-time prognostics for internet-of-things (IoT) applications in the fields of manufacturing, transportation, and utilities. For such applications, we have achieved greatest prognostic success with ML algorithms from a class of pattern recognition known as nonlinear, nonparametric regression. To intercompare candidate ML algorithmics to identify the "best" algorithms for IoT prognostic applications, we use three quantitative performance metrics: false alarm probability (FAP), missed alarm probability (MAP), and overhead compute cost (CC) for real-time surveillance. This paper presents a comprehensive framework, SimML, for systematic parametric evaluation of statistical ML algorithmics for IoT prognostic applications. SimML evaluates quantitative FAP, MAP, and CC performance as a parametric function of input signals' degree of cross-correlation, signal-to-noise ratio, number of input signals, sampling rates for the input signals, and number of training vectors selected for training. Output from SimML is provided in the form of 3D response surfaces for the performance metrics that are essential for comparing candidate ML algorithms in precise, quantitative terms.**

Keywords—Machine Learning; IoT Prognostics; AAKR; MSET

I. INTRODUCTION

I.1 Background: Electronic Prognostics (EP) Algorithmic Innovations Adapted for IoT Prognostics

New prognostic algorithms are being adapted and optimized for real-time Internet-of-Things (IoT) applications across a variety of IoT customer industrial applications including Manufacturing, Utilities, and Transportation.

Oracle has over the last 15 years developed and patented a suite of advanced statistical pattern recognition innovations for enterprise computing components, subsystems, and for integrated hardware-software systems in enterprise and cloud data centers. As is the case for large-scale IoT industrial applications, enterprise computing prognostics face challenges with very large collections of sensors from expensive assets. A typical enterprise server today contains hundreds of physical transducers (4RU server), and up to 3400 transducers for a rack-sized engineered system. A medium size enterprise data center comprises over 1M sensors streaming digitized time-series signatures on a 24x7x365 basis. In this paper we adapt large-scale dense-sensor prognostic algorithmics for IoT industrial applications and present a comprehensive pluggable framework, called SimML, for rapidly customizing, tuning, optimizing, and validating candidate prognostic machine learning (ML) algorithms for industrial IoT applications where one needs to rigorously validate that prognostic functional requirements are being met. Moreover, SimML allows quantitative intercomparison of candidate ML prognostic algorithms in terms of what we have found to be the most important prognostic functional requirements for business critical and mission critical IoT applications: the false alarm probability (FAP), missed alarm probability (MAP), and the overhead compute cost (CC) for real time prognostic surveillance. We begin with a brief background on the extensive portfolio of prognostic applications we have adopted from Oracle's mature portfolio of enterprise data center prognostic applications to dense-sensor IoT applications.

Electronic Prognostics (EP) for business-critical and mission-critical IT systems comprises a comprehensive methodology for proactively detecting and isolating failures, recommending condition-based maintenance (CBM), and estimating in real time the remaining useful life (RUL) of critical components. The key enabler for achieving Electronic Prognostics capabilities is Oracle's continuous system telemetry harness (CSTH) [Ref. 1], which collects and preprocesses any/all types of time series signals relating to the health of dynamically executing components and subsystems. These time series signatures provide quantitative metrics associated with physical variables (a typical data center now contains > one million physical sensors inside the IT assets measuring distributed temperatures, voltages, and currents, power metrics, fan speeds, vibration sensors), performance variables (CPU & memory loads, throughputs, queue lengths, process metrics, etc.), and various quality-of-service (QOS) performance metrics. The CSTH signals are continuously archived to an offline circular file (i.e. the "Black Box Flight Recorder"), and are also processed in real time using advanced statistical ML algorithms for proactive anomaly detection and for RUL estimation with associated quantitative confidence factors. Oracle has achieved highest prognostic performance with lowest overhead compute cost by leveraging ML techniques from the class of mathematics known as nonlinear, nonparametric (NLNP) regression. Oracle's continuous system telemetry harness (CSTH) coupled with NLNP ML pattern recognition [Refs 2-6] help to increase component reliability margins and system availability goals while reducing (through improved root cause analysis) costly sources of "no trouble found" (NTF) events from spurious false alarms that cause down time in customer's critical assets.

Proactive fault monitoring is the ability to identify leading indicators of failure before the failure actually occurs. Empirical ML pattern recognition methods are frequently used in proactive fault monitoring, whereby the signal behaviors are modeled from telemetry signals collected during normal operation with undegraded assets. The pattern recognition model is constructed in a training phase, during which the (nonlinear) correlations among the input signals are learned. In the monitoring phase, the pattern recognition ML module is used to estimate the value of each signal as a function of the other signals. Significant deviations between the estimates and observed signals indicate a potential incipient degradation mode in the system. This paper reports a systematic framework for evaluation, tuning, optimization, and evaluation of candidate ML algorithms using real and synthesized data streams.

The three most important evaluation criteria that make or break ML algorithms for end customer applications are the FAP, MAP, and ComputeCost. If any one or more of these 3 metrics fails to meet Prognostic Functional Requirements (PFRs), the ML technique can be useless (at best) and dangerous (for customers with safety-critical assets) at worst.

A related functional requirement adapted from EP prognostic applications for IoT [Refs 7-9] is that the FAP and MAP have to be separately configurable. If one evaluates a candidate ML algorithm for which the "tuning" parameters cause the MAP to go up when one lowers the FAP, or vice versa, then the ML algorithm will not be useful for real-time prognostic applications.

We approach the PFR evaluation with a systematic Design-of-Experiments (DOE) framework that evaluates the 3 primary functional requirements parametrically as a function of the following important variables for any IoT prognostic challenge:

* Number of signals available
* Sampling rate of signals
* Degree of cross correlation among available signals
* Signal-to-noise ratio for individual signals
* Number of training vectors selected for prognostic model training
* "Degree of severity" for anomalies one hopes to detect with the ML prognostics (SimML approaches this systematically and parametrically, by "analytically" dialing down the anomaly signatures to generate [FAP,MAP,ComputeCost] curves vs subtleness of injected anomalies.
* Influence of lead/lag relationships between/among monitored signals. [This often "invisible" aspect of time-series data sets will invalidate evaluations of most types of ML algorithms. There are very many reasons that time-varying leads/lags get into time series datasets, whether from business applications, industrial IoT applications in manufacturing, utilities, transportation, military systems, and all types of enterprise computing and IoT cloud data center applications. Most prognostic ML algorithms will give drastically varying results when there are time-varying leads/lags inherent in the testing datasets.]

Finally, one of the most important aspects to be aware of when evaluating candidate ML algorithms for prognostic applications is an essential functional requirement we call "robustness". Many naive approaches to ML prognostics research attempt to evaluate a ML algorithm by "how well it predicts" new observations, after being trained on historical observations. Injudicious reliance on that criteria is dangerous for the following reason: If the dataset of time series signals is a "good" data set in the sense that the signals really possess a good degree of inter-correlation, then it is correct that a good ML technique can be trained and can predict new observations on the basis of the other correlated variables, and in this case anomalies can be detected with high accuracy with the techniques described in this paper. However, if a ML data scientist accidentally gets a data set where there is no correlation among the signals, many ML algorithms will "learn the noise" and seem like they are predicting well, but will have zero capability for detecting any real anomalies. Our "Robustness" measurement achieves the following: If a small disturbance is injected into signal #N, then Alerts are generated immediately for signal #N and for no other signals that do not have a disturbance. Robustness needs to be an essential element of any framework for evaluation of candidate ML algorithms for IoT prognostic applications.

I.2 Sequential Probability Ratio Test (SPRT) for Fault Detection

A prognostic technique based upon empirical NLNP regression coupled with a SPRT for fault annunciation provides a superior surveillance tool because it is sensitive not only to disturbances in signal mean, but also to very subtle changes in the statistical moments (mean, variance, skewness, kurtosis) of the monitored signals and the patterns of correlation between/among multiple types of signals. Our prognostic algorithms adapted from EP to IoT prognostic challenges employ a statistical pattern recognition technique called the Sequential Probability Ratio Test (SPRT) [Refs 9-13], which provides the basis for detecting very subtle statistical anomalies in noisy process signals at the earliest mathematically possible time, thereby providing actionable warning-alert information on the type and the exact time of onset of the disturbance. Instead of simple threshold limits that trigger faults when a signal increases beyond some threshold value, the SPRT technique is based on user-specified false-alarm and missed-alarm probabilities (FAPs and MAPs), allowing the end user to control the likelihood of missed detections or false alarms. For sudden, gross failures of sensors or IoT system components the SPRT annunciates the disturbance as fast as a conventional threshold limit check. However, for slow degradation that evolves over a long time period (gradual decalibration bias in a sensor; very subtle voltage drift from the variety of aging mechanisms that cause resistances to change very slowly with age; bearing out-of-roundness degradation, lubrication dryout, shaft centerline eccentricities, rotator-vane imperfections, or buildup of a radial rub in all types of rotating machinery and centrifugal pumps; the gradual appearance of new vibration spectral components in the presence of noisy background signals, etc.), the SPRT raises a warning of the incipience or onset of the disturbance long before it would be apparent to any conventional threshold based rules.

II. Example Results for SimML Monte Carlo Simulation Computations with Two Candidate ML Algorithms

Example computations with SimML are presented for intercomparison of two prognostic ML algorithms, both of which are integrated with a SPRT "detector" algorithm for annunciation of anomalies in noisy process variables. The first candidate technique is Auto Associative Kernel Regression (AAKR), as developed by our collaborating partner prognostics researchers at the U. of Tennessee [Refs 11,14]. The 2^{nd} candidate technique is the Multivariate State Estimation Technique (MSET) [Refs 15-19]. SimML is employed to conduct a parametric prognostic accuracy and compute-cost analysis for these two candidate algorithms using identical signal data sets for the corresponding runs, and while varying through all possible permutations and combinations of the number of signals under surveillance, the number of observations (and hence the sampling rates for the signals), the degree of cross correlation among the monitored signals, and the signal-to-noise ratio for the individual signals (which we systematically vary by superimposing noise onto otherwise well-correlated signals…to simulate increasingly challenging cases where the correlation content…which empirical ML algorithmics thrive on, becomes increasingly obscured by "background noise").

Fig 1. Shows just one (of hundreds) of data sets we employ for this exhaustive design-of-experiments analysis. In this case, we begin with 12 monitored signals, 11 of which are "real" process variables with a moderate degree of cross correlation (correlation coefficients slightly greater than .6), and a 12^{th} signal that is purely random.

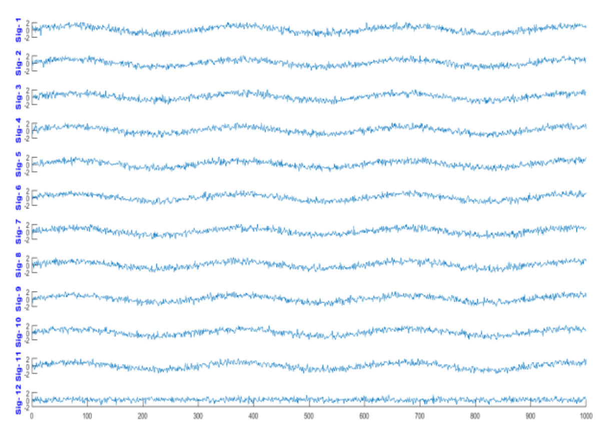

Fig. 1. Example Signal Dataset for Analysis: 11 Signals with Process Correlation + 1 random signal

The reason we include one random signal to each data set under analysis is twofold: (1) in very many types of industrial prognostic applications the sensors have shorter mean-time-between-failure (MTBFs) than the assets being monitored. It can commonly be the case that a sensor may fail in service and go unnoticed. Prognostic pattern recognition algorithms need to be robust to individual sensor failures; and (2) if a human data scientist accidentally includes a signal in the surveillance framework that is not correlated with any other signals,that "dilutes" the prognostic accuracy for the remaining process signals. Of course we know that even crude prognostic algorithms will do outstanding if fed extremely well correlated signals. In practice, a ML researcher is lucky to get clusters of signals with correlation coefficients > .5, and with high resolution (high signal-to-noise ratios) for the individual signals. For this evaluation we make sure to include increasingly challenging scenarios wherein we systematically diminish the degree of correlation between/among monitored signals, and we increase the noise contamination on the signals to explore accuracy, false-alarm avoidance, and overall compute cost under these increasingly challenging scenarios.

Fig. 2 shows a typical output with a model estimated signal (in this case with AAKR) superimposed on the monitored signal. Both AAKR and MSET do quite well predicting the correlated components in the monitored process signals. With only 50 training vectors (out of 1000 observations), we can see that the "residuals" (difference between the measured signal and the predictions) are quite small (Fig. 3). We also see in Fig. 3 there are no false alarms from the SPRT Detector while monitoring this process which has no anomalies present, as desired. That is the case for both AAKR and MSET for the example data set shown.

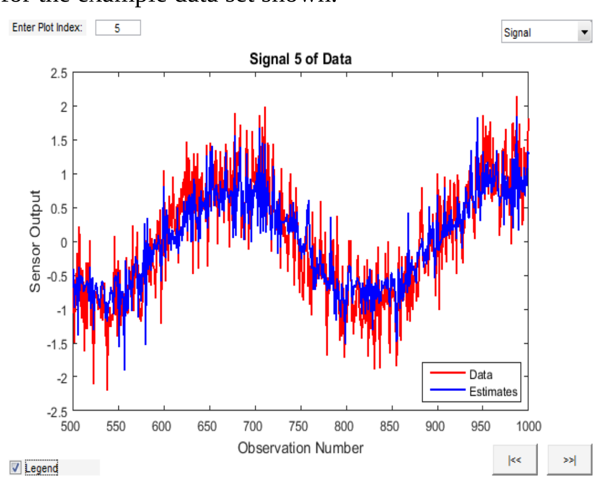

Fig. 2. AAKR Estimates: 12 signals, 50 training vectors

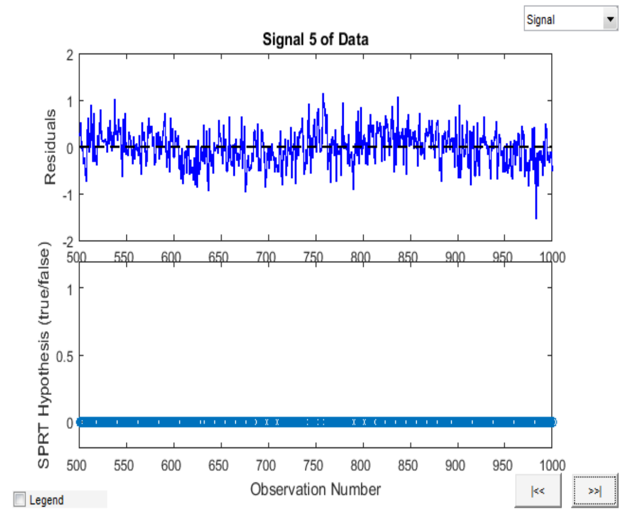

Fig. 3. AAKR SPRT: H=1, 12 signals, 50 training vectors

One challenge we encounter with conventional AAKR is that the "kernel bandwidth parameter," H, needs to be selected judiciously to obtain good predictions with AAKR. If H is not optimized for every data set and every signal in the data set, predictions and false-alarm performance can be quite degraded. This is illustrated in Fig. 4 where we plot the prediction uncertainty (in root-mean-square-error, RMSE) for the AAKR predictions vs H. Although prognostic performance is good when H is optimized, this optimization step increases the compute cost for AAKR applications. By contrast, MSET does not need any tuning optimizations.

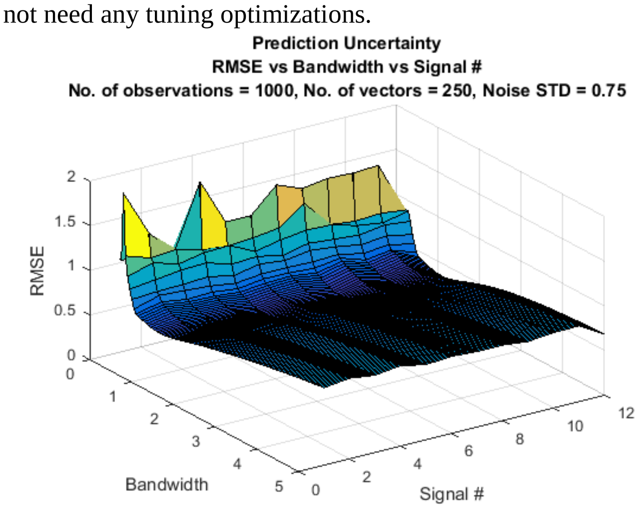

Fig. 4. Kernel Bandwidth Optimization for AAKR

For the hundreds of cases analyzed as part of this investigation, MSET consistently gives significantly higher prognostic accuracy Fig. 5 shows the accuracy of estimates from AAKR as a function of the standard deviation of the analyzed signals and the number of training vectors selected for modeling. The corresponding accuracy as a function of the same control variables is shown in Fig. 6.

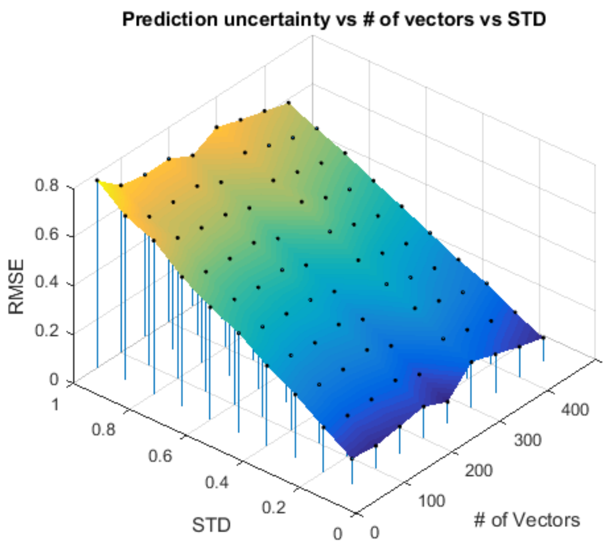

Fig. 5 AAKR Prediction Uncertainty vs Noise Contamination on Signals vs # of Training Vectors Selected

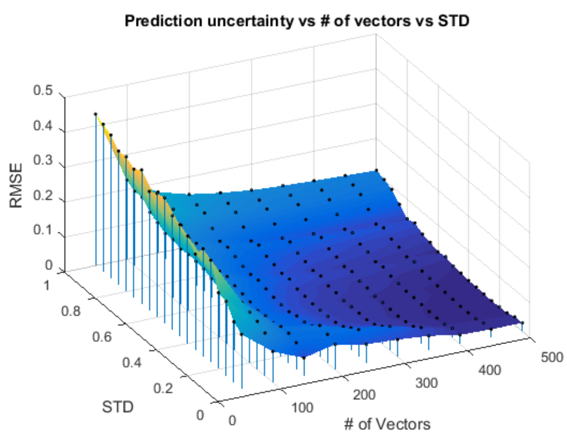

Fig.6 MSET Prediction Uncertainty vs Noise Contamination on Signals vs # of Training Vectors

Fig. 7 shows the compute cost for AAKR (implemented in Java) for test analyses where the number of signals, and the number of observations per signal are varied. The corresponding results for MSET (also implemented in Java) are shown in Fig. 8.

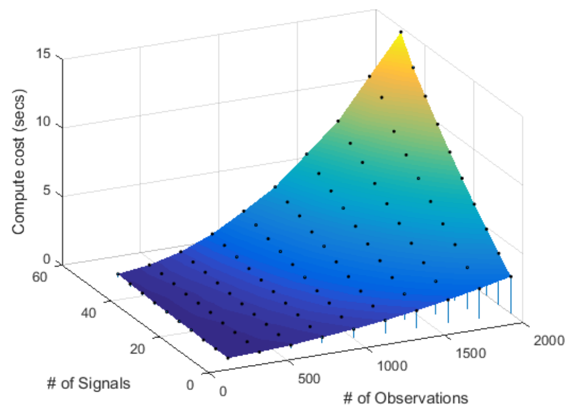

Fig. 7 AAKR Compute Cost versus Number of Signals and Number of Observations per Signal

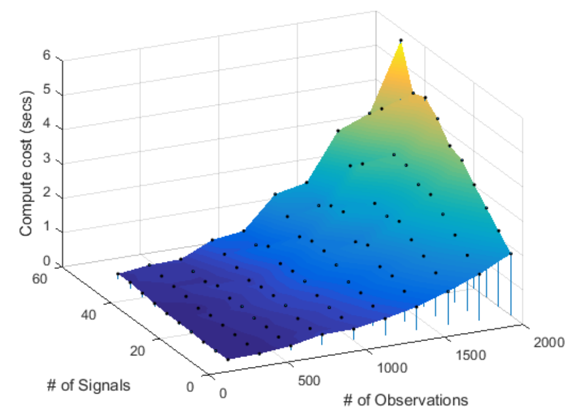

Fig. 8 MSET Compute Cost versus Number of Signals and Number of Observations per Signal

CONCLUSIONS

Advanced statistical machine learning (ML) algorithms are being developed, trained, tuned, optimized, and validated for real-time prognostics for internet-of-things (IoT) applications in the fields of manufacturing, transportation, and utilities. Greatest prognostic success to date has been achieved with ML algorithms from a class of pattern recognition known as nonlinear, nonparametric (NLNP) regression. A comprehensive framework, SimML, has been developed for systematic parametric evaluation of statistical ML algorithmics for IoT prognostic applications. SimML evaluates quantitative FAP, MAP, and CC performance as a parametric function of input signals' degree of cross-correlation, signal-to-noise ratio, number of input signals, sampling rates for the input signals, and number of training vectors available for training. Output from SimML is provided in the form of 3D response surfaces for the performance metrics that are essential for comparing candidate ML algorithms in precise, quantitative terms. For any candidate ML algorithms coded in Matlab or in Java, SimML employs a nested-loop structure to enable a comprehensive design-of-experiments evaluation of FAP, MAP, and CC as a function of the dependent variables and computes 3D response maps for all pair-wise combinations if dependent variables. Examples have been presented in this paper for two candidate ML prognostic algorithms: AAKR and MSET. Results have shown that MSET achieves substantially better prognostic performance, higher accuracy for signal predictions, higher sensitivity for detection of subtle anomalies in noisy process variables, lower FAP and MAP, and significantly lower compute cost (approximately 80% lower) for real time surveillance applications.

REFERENCES

[1] "Prognostics of Electronic Components: Health Monitoring, Failure Prediction, Time To Failure," K. G. Gross, K. W. Whisnant and A. M. Urmanov, ***Proc. New Challenges in Aerospace Technology and Maintenance Conf. 2006***, Suntec City, Singapore (Feb 2006).

[2] "Electronic Prognostics Techniques for Mission Critical Electronic Components and Subsystems," K. C. Gross, K. W. Whisnant and A. M. Urmanov, ***Proc. 2006 Components for Military and Space Electronics Symposium,*** Los Angeles, CA, (Feb 2006).

[3] "Improved Methods for Early Fault Detection in Enterprise Computing Servers," K. C. Gross and K. Mishra, 2004 SAS Users Group International (SUGI 29), Montreal, Canada. (May 9 – 12, 2004).

[4] "MSET Performance Optimization for Proactive Detection of Software Aging," K. Vaidyanathan and K. C. Gross, Proc. 14th IEEE Intn'l. Symp. on Software Reliability Eng. (ISSRE'03), Denver, CO (Nov. 2003).

[5] K. J. Cassidy, K. C. Gross, and A. Malekpour, "Advanced pattern recognition for detection of complex software aging phenomena in online transaction processing servers," in *Proc. Intnl. Performance and Dependability Symposium*, Washington, D.C., June 23 - 26, 2002.

[6] K. Vaidyanathan and K. C. Gross, "Proactive detection of software anomalies through MSET," *Proc. IEEE Workshop on Predictive Software Models (PSM-2004)*, Chicago, Sept 17-19, 2004.

[7] K. C. Gross, K. Vaidyanathan, A. Bougaev, and A. Urmanov, "Round-Robin Staggered-Imputation (RRSI) Algorithm for Enhanced Real-Time Prognostics for Dense-Sensor IoT Applications," ***International Conference on Internet Computing and Internet of Things (ICOMP'16)***, Las Vegas, NV (July 25-28, 2016).

[8] K. C. Gross, K. Vaidyanathan, and M. Valiollahzadeh, "Advanced Pattern Recognition for Optimal Bandwidth and Power Utilization for Intelligent Wireless Motes for IoT Applications," ***17th International Conference on Wireless Networks (ICWN'16)***, Las Vegas, NV (July 25-28, 2016).

[9] T. Masoumi and K. C. Gross, "SimSPRT-II: Monte Carlo Simulation of Sequential Probability Ratio Test Algorithms for Optimal Prognostic Performance," 2016 International Symposium on Artificial Intelligence (CSCI-ISAI), Las Vegas, NV (Dec 15-17, 2016).

[10] A. Wald, *A Sequential Analysis*, New York: Wiley, 1947.

[11] B. A. Jeffries, J. W, Hines, and K. C. Gross, "Intrusion Detection of a Simulated SCADA System using Data-Driven Modeling," 38th IEEE Symposium on Security and Privacy (SP2017), San Jose, CA, (May 22-24, 2017).

[12] Multivariate SPRT for Improved Electronic Prognostics of Enterprise Computing Systems," K. C. Gross and R. Dhanekula, Proc. 65th Meeting of the Machinery Failure Prevention Technology Society (MFPT2012), Dayton, OH (April 2012).

[13] "Novel Training Enhancements for Advanced Statistical Pattern Recognition Used for Electronic Prognostics of Enterprise Computing Systems," K. C. Gross, R. Dhanekula, and K. Vaidyanathan, Proc. IEEE World Congress in Computer Science, Computer Engineering, and Applied Computing (WorldComp2011), Las Vegas, NV (Aug 2011).

[14] D. Garvey and J W. Hines, "The Development of a Process and Equipment Monitoring (PEM) Toolbox and its Application to Sensor Calibration Monitoring", *Quality and Reliability Engineering International*, **22**, 1-13, 2007.

[15] R. Singer, K. C. Gross, J. Herzog, R. King, S. Wegerich, "Model-based nuclear power plant monitoring and fault detection: Theoretical foundations," in *Proceedings from the Intelligent System Application to Power Systems Conference*, pp. 60-65, July 1997.

[16] K. C. Gross, S. Wegerich, and R. M. Singer, "New artificial intelligence technique detects instrument faults early," *Power Magazine*, vol. 42, no. 6, pp. 89-95, 1998.

[17] K. C. Gross, R. M. Singer, S. W. Wegerich, J. P. Herzog, R. VanAlstine, and F. Bockhorst, "Application of a Model-based Fault Detection System to Nuclear Plant Signals," Proc. 9th Intnl. Conf. On Intelligent Systems Applications to Power Systems, pp. 66-70, Seoul, Korea (July 6-10, 1997).

[18] "Fault-Tolerance Improvement for a MSET Model of the Crystal River-3 Feedwater Flow System," A. Miron, S. Wegerich, F. Yue, K. C. Gross, and J. Christenson, ***Proc. IEEE Nuclear Science Symp. and Medical Imaging Conf.,*** Toronto (Nov 1998).

[19] "Multivariate State Estimation Technique (MSET) Surveillance System," J. P. Herzog, K. C. Gross, S. W. Wegerich, and R. M. Singer, Appendix H of ***On-Line Monitoring of Instrument Channel Performance***, **TR-104965**, EPRI (Oct 1998).