INL/EXT-19-55407
Revision 0
ORNL/SPR-2019/1310
Revision 0

Light Water Reactor Sustainability Program

Determination of Sensor Quality of Calibration Using Advanced Data Analytics and Machine Learning Methods

September 2019

U.S. Department of Energy

Office of Nuclear Energy

INL/EXT-19-55407
Revision 0
ORNL/SPR-2019/1310
Revision 0

Determination of Sensor Quality of Calibration Using Advanced Data Analytics and Machine Learning Methods

Sacit M. Cetiner, Nageswara Rao, Pradeep Ramuhalli, Christopher Greulich, Pravallika Devineni
Oak Ridge National Laboratory

Fan Zhang
University of Tennessee

Anil Gurgen
North Carolina State University

Vivek Agarwal
Idaho National Laboratory

September 2019

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

ABSTRACT

Manual sensor calibration activities in light water reactor (LWR) operations are labor-intensive and constitute a significant amount of the high operations and maintenance costs, impeding the economic competitiveness of nuclear power. Online monitoring (OLM) of sensor calibrations using advanced data analytics and machine learning techniques may help to improve the economics of nuclear power while maintaining the safety and reliability of operations by focusing recalibrations on necessary periods. However, challenges associated with uncertainties in sensor drift detection and quantification have limited the use of OLM in the commercial nuclear power fleet.

This report describes research leveraging advances in data analytics and machine learning methods to address technical challenges in sensor drift detection and uncertainty quantification. Specifically, the report presents results using the Auto-Associative Kernel Regression (AAKR) and principal component regression (PCR) methods that have been studied over the years, as well as hetero-associative information fusion methods using ensemble of trees (EOT) and support-vector machine (SVM) methods proposed under this project. These methods are compared using data from laboratory measurements, as well as limited data from instrumented irradiation tests. The results indicate the potential for drift detection and quantification as a precursor to identifying sensors that are out of calibration. Drift detection and quantification is also applied to identify failing sensors. In all cases, the proposed methods appear to compare well against existing methods (AAKR and PCR) by several measures. They also appear to be more accurate in estimating drifts in sensor measurements, indicating a potential application for early detection of deviations in sensor responses.

EXECUTIVE SUMMARY

In the operations of light water reactor (LWR) nuclear plants, sensor calibration is a labor-intensive activity. Experience shows that most sensors remain within calibration, and a purely time-based, prescriptive recalibration of sensors can add unnecessary cost and time to outage schedules. Furthermore, unnecessary recalibrations increase the risk of unintended damage to sensors through human error.

Online monitoring (OLM) of sensor calibrations using advanced data analytics and machine learning techniques may help eliminate many of these inefficiencies by focusing recalibration efforts only where they are necessary. While OLM has been the subject of multiple research and development efforts over the past 25+ years, challenges remain in uncertainty quantification, sensor drift detection, and estimation. These issues have limited the use of OLM in the commercial fleet. Recent advances in anomaly detection, sensor fusion, and uncertainty quantification methods, along with a focus on improving the economic competitiveness of nuclear power, make adoption of OLM more likely.

The research presented in this report focuses on leveraging advances in data analytics and machine learning methods to address technical challenges in sensor drift detection and uncertainty quantification. The outcome of this research will (1) advance the state of technology for monitoring tools used in online sensor calibration, (2) establish a foundation to implement robust, accurate online sensor calibration at a pilot nuclear utility in fiscal year 2020, and (3) help nuclear power plants (NPPs) to minimize inefficiencies in preventive maintenance and enhance cost savings.

The overall objective of this work is to develop a robust methodology to address these issues using machine learning and data analytics. The methodology will automatically identify symptoms due to sensor and instrument calibration drifts that exceed the specification limits, as well as those indicating complete sensor failures. This report focuses on detecting and quantifying sensor drifts caused by calibration changes and conditions such as sensing line blockage or leaks (failure modes of pressure transmitters) to a limited extent. The primary objective is to generate uncertainty quantification (UQ) estimates that are improved by considering the information specific to the underlying NPP systems.

This report examines existing auto-associative methods and proposes hetero-associative methods for building models of the sensor response using measurement data from normal operations. Auto-Associative Kernel Regression (AAKR) and principal component regression (PCR) methods are evaluated, along with information fusion methods developed under this project using ensemble of trees (EOT) and support vector machine (SVM) methods. All techniques use available training measurements to develop empirical models that implicitly incorporate the relationships between measurements from different sensors. AAKR and PCR provide a baseline set of results for comparison to determine performance gains from other information fusion methods. Complementary approaches using first-principles models to inform the relationship between different measurements are not examined in this work.

Available data sets include data from laboratory flow-loops in which sensor drift from calibration changes or sensor faults was artificially introduced. Data from instrumented irradiation tests were also available but did not have enough ground truth information sensor drifts with magnitudes and durations. However, initial analyses appear to indicate the ability to detect deviations from normal sensor behavior as indicated by measurements consistent with most redundant sensors.

Application of the algorithms to these data sets indicates that the proposed methods compare favorably against standard OLM algorithms (AAKR and PCR) on accuracy and sensitivity overall. Furthermore, the algorithms provide more accurate estimates under large drift conditions. However, these results need to be verified using field data sets from NPP operations. Access to such data is being obtained and will lead to future work to further advance this technology.

In addition to the development of techniques to detect and quantify drift in sensor responses, the generalization equations for the predicted model output were developed. These equations are based on statistical theory, and they establish that the underlying problem is solvable in principle. The equations result in reasonably tight bounds for the data sets considered. Improving the tightness of the error bounds requires sharpening the generalization equations by incorporating specific parameters of the underling system and those of the EOT and SVM methods.

Ongoing research includes evaluating additional data sets from test reactors and plant operational data. Approaches for using the algorithm outputs for recalibrating the sensor online will be examined next. Approaches to address regulatory concerns to accelerate deployment of OLM in the fleet are being evaluated under a different research program by other organizations. This project will engage with these organizations to better understand technical and regulatory barriers to deployment and will update the research plan to address these barriers in the next phase.

ACKNOWLEDGEMENTS

This report was made possible through funding by the United States Department of Energy's Light Water Reactor Sustainability Program. We are grateful to Alison Hahn of the United States Department of Energy, and Bruce Hallbert and Craig A. Primer at Idaho National Laboratory for championing this effort. We also thank Barney C. Hadden and Jodi L. Vollmer at Idaho National Laboratory for technical editing and formatting of this report.

CONTENTS

FIGURES

TABLES

ACRONYMS

AAKR	Auto-Associative Kernel Regression
AGR	Advanced Gas Reactor
ATR	Advanced Test Reactor
CDF	core damage frequency
DOE	US Department of Energy
DP	differential pressure
EOT	ensemble of trees
EPRI	Electric Power Research Institute
ISA	International Society for Automation
LAR	license-amendment request
LERF	large early release frequency
LWR	light water reactor
MSE	mean square error
NE	Office of Nuclear Energy
NLPCA	nonlinear principal component analysis
NPP	nuclear power plant
NRC	US Nuclear Regulatory Commission
OLM	online monitoring
PC	principal component
PCA	principal component analysis
PCR	principal component regression
RMS	root mean square
RMSE	root mean square error
ROC	receiver operating curve
RP	recommended practice
RTD	resistance temperature device
SER	safety evaluation report
SPRT	sequential probability ratio test
SVM	support-vector machine

TC	thermocouple
TS	technical specification
UQ	uncertainty quantification
US	United States
VC	Vapnik-Chervonenkis

1. INTRODUCTION

In the operations of light water reactor (LWR) nuclear power plants (NPPs), time-based, prescriptive-maintenance activities typically include scheduled maintenance, inspection, calibration, and replacement of sensors and transmitters. Sensor calibration is a labor-intensive activity that may not always be necessary. Experience indicates that most sensors remain within calibration, and consequently, purely time-based, prescriptive recalibration of sensors can add unnecessary cost and time to outage schedules. Furthermore, such unnecessary recalibrations increase the risk of unintended damage to sensors through human error.

Online monitoring (OLM) of sensor calibrations using advanced data analytics and machine learning techniques may help to eliminate many of these inefficiencies. OLM techniques have been available for several years and are the subject of US Nuclear Regulatory Commission (NRC) safety evaluation report (SER) [1]. However, no US utility has implemented OLM to date for the purpose of extending the calibration-intervals. The only routine implementation of OLM technology for monitoring to extend calibration intervals appears to be at the Sizewell B NPP in the United Kingdom [2,3].

The NRC identified several questions [1] that should be addressed before OLM is implemented by a licensee. These questions cover a broad range of topics, including:

- Uncertainty quantification (UQ)
- Single-point monitoring and sensor failure modes
- Plant-specific applicability

The research presented in this report focuses on methods to detect and estimate sensor measurement drift, as well as analysis of their generalization errors. Sensor drift is characterized by gradual changes in sensor measurements while the underlying plant parameters are maintained within normal operational ranges. Drift due to calibration changes typically manifests itself gradually in sensor measurements. However abrupt sensor response changes can also be caused by factors such as sensor failure or electromagnetic interference.

The overall objective of this effort is to develop a robust methodology to address these issues using machine learning and data analytics methods. The method would automatically identify symptoms due to sensor and instrument-calibration drift that exceed the allowable specification limits, as well as symptoms that indicate sensor failures.

This report described data analytics techniques, specifically information fusion methods using multiple sensor measurements, to estimate sensor drift. Results are presented using Auto-Associative Kernel Regression (AAKR) and principal component regression (PCR) that have been studied over the years, as well as hetero-associative information fusion methods using ensemble of trees (EOT) and support vector machine (SVM) methods that were developed under this project. These new techniques use available training measurements to develop empirical models that implicitly incorporate the relationships between measurements from different sensors. If the empirical model prediction errors are well-bounded by empirical estimates and analytical means, then the models can be applied in an online calibration process. The performance of these methods is tested using available test measurements to detect and quantify the sensor-drift behavior. Complementary approaches that use first-principles models of the process to inform the relationship between different measurements are not examined in this work.

The outcome of this research will (1) advance the state of technology for monitoring tools used in online sensor calibration, (2) establish a foundation for robust and accurate online sensor-calibration implementation at a pilot nuclear utility in fiscal year 2020, and (3) help NPPs to minimize inefficiencies in the current preventive-maintenance strategy and to enhance cost savings.

2. BACKGROUND

This section provides a brief overview of OLM in the nuclear industry and outlines existing research addressing technical gaps. A detailed review of OLM in the nuclear industry is available in the literature [4].

Analog sensors and transmitters have been used from the early days in the commercial nuclear industry for measuring key parameters in the primary and secondary loops (parameters such as temperature, flow, level, pressure, and neutron flux) [5]. These sensors provide continuous measurements of a quantity within their operating range and, in conjunction with the transmitter electronics, provide a standard current output (usually 4–20 mA) for use in subsequent analysis and decision logic. Typically, the current signal is sent to a current-to-voltage converter and then to instrument racks for processing, recording, and display. Recent advances in measurement technology has led to digital transmitters that have an analog front end with microprocessors and other logic circuits. The output of these digital transmitters is a digital signal which is used in the subsequent analysis, display, and recording of key measurement parameters.

Safety-related sensors are required to periodically have their calibration checked and adjusted as necessary to maintain sensor calibration within a prescribed tolerance [4, 6]. Calibration typically involves exposing the sensor to one or more known inputs and adjusting the readout so that it matches the known value. Current industry practice for maintenance of safety and non-safety sensors requires removal, recalibration, and reinstallation of every sensor and its associated channels [7]. Note that sensor calibration checks can include the response of the sensor and the response time, although some studies indicate the potential to identify response time changes based on the sensor output for specific types of sensors [8].

Sensor aging and degradation leads to changes in calibration that usually manifest themselves as a drift in the sensor signal. In addition, sensors can fail in several modes resulting in changes in the measured response or response time; the specific modes depend on the type of sensor and the environment in which it is installed and used. Failure modes can include failure of the sensor head alone or failure of the instrumentation (e.g., the electronics in racks or cabinets). Damage to the instrumentation cabling can also be a problem and will usually result in added noise and, possibly, drift in analog signals.

While periodic assessments of sensor calibration can detect and correct for sensor aging or failure, approaches that continuously monitor sensor response to identify those that are out of calibration are attractive from the perspective of reducing the cost of maintenance. These OLM methods often use measurements from a reliable set of sensors to build a model. The model is then used for anomaly detection. Cross calibration can also be applied to detect drift in a redundant set of sensors such as resistance temperature devices (RTDs). Guidance on acceptable application of cross-calibration for RTDs is provided in NUREG 0800 BTP 7-13 [9].

2.1 Prior Research in OLM

The motivation for OLM is to avoid recalibration of sensors every 18–24 months. This recalibration process is costly, and disconnection/reconnection of sensors for periodic calibration can introduce additional errors. OLM is now being considered to extend surveillance frequency. The objective is to delay unnecessary functional or calibration checks for sensors and instrumentation beyond the intervals specified in the technical specification (TS). If it is possible to delay recalibration without significantly increasing risk metrics such as core-damage frequency (CDF) and large early release frequency (LERF), maintenance costs would be reduced by allowing plants to defer unnecessary test and maintenance actions.

Techniques for OLM were developed and tested in the late 1990s, and an Electric Power Research Institute (EPRI) topical report was submitted to the NRC for evaluation [10]. The NRC SER on this topical report concludes that licensees could use OLM for calibration interval extension as long as several requirements are met [1,11]. While US utilities indicated interest in pursuing OLM for calibration interval extension [12,13], no license amendment applications have been made by US utilities for OLM use.

Sizewell B in the United Kingdom adopted OLM to extend intervals between calibrations. They applied drift analysis to historical data to demonstrate low drift from the sensors. OLM was applied to monitor performance to identify sensor drift during extended intervals. In this type of scenario, the OLM approach ensures that any unexpected failures or drift in the measurements can be detected in a reasonable time frame. As part of program to extend intervals between calibrations, one in four channels undergoes calibration during every outage. This ensures that all channels are calibrated at least once in an eight-year cycle [14,15,16].

Recent research findings have led to proposals to eliminate response-time testing, relying on TS surveillance testing to detect conditions that would affect response time [17]. However, it is not clear how extending intervals between calibrations will impact the need to monitor response time. It is possible that using OLM to monitor sensor calibration may be applied to monitoring both sensor response and response time [18] if the associated uncertainty can be properly characterized.

2.2 Open Questions and Focus of This Work

Several questions and considerations need to be answered regarding the use of OLM to extend intervals between calibrations. While these issues will not impede NPP licensee implementation, technical advances to address these questions may help demonstrate that using OLM for this purpose would result in a minimal increase in risk Open questions are presented and discussed below.

1. **Uncertainty quantification in sensor drift estimation.** Given the data used to develop OLM models, the generalization error associated with model prediction must be quantified. Several approaches can be used, including International Society for Automation (ISA) Recommended Practice (RP) 67.04.02, "Methodologies for the Determination of Setpoints for Nuclear Safety-Related Instrumentation," to characterize the uncertainty in setpoint determination and the uncertainty associated with OLM model predictions using future test data (generalization error). However, these methods tend to be conservative in their estimates of model prediction error, and the large UQ estimates can drive down the operating margins to levels not useful in practice. Tighter, more practical uncertainty estimates are needed. The primary objective in addressing this question is to generate improved UQ estimates by considering the information specific to the underlying NPP systems. Using the data available under normal or known drift conditions, models can be trained to detect and estimate sensor drifts. The training error is computed and used as a component of generalization error, which depends on the properties and parameters of the method.
2. **Fault detection.** Most OLM methods are applied with the NPP operating under steady-state conditions and only monitor the current response from the sensor. Detecting drift in sensor responses under different operating conditions is a challenge. This is more so when sensor failures occur with corresponding changes in the recorded responses. Signatures from sensor data that indicate sensor failures need to be identified. While drift in the data is one indication of potential failure, other signatures may also help identify the presence of sensor failure. This effort is focused on detecting and quantifying sensor drifts caused by calibration changes and other conditions such as sensing line blockage or leaks (some failure modes of pressure transmitters) to a limited extent.

3. **Online recalibration and virtual sensing.** These challenges are associated with correcting a detected sensor drift online without resorting to manual recalibration. If sensor drift—either incipient or after some time—has been detected, a correction factor can be calculated for recalibration so that this single-point calibration does not degrade the calibration at other points in the span of the sensor. The sensor measurement can also be replaced with a calculated value (analytic or virtual sensor), but the generalization error bounds for this calculation must also be quantified as well. The assumption is that carrying forward a calculated value results in a biased estimate from the virtual sensor that no longer reflects the true state of the reactor, and this divergence changes with time.

From the questions and considerations presented above, a few key research topic needs are identified:

- **Analysis techniques for detecting sensor drift in real time.** These techniques may include feature extraction, machine learning, model building, etc., but they be data driven. Ideally, the techniques will detect incipient faults. These methods should be characterized in terms of common metrics like detection and false-alarm rates and the operating point in the receiver operating curve (ROC)—metrics often used in established detection theory [19,20].
- **Analysis techniques for drift detection that exploit the knowledge of sensor failure modes and failure physics, including those leading to calibration changes.** These techniques are likely to be challenging, but they may be needed if other simpler approaches are deemed inadequate.
- **Methodology for online recalibration of a drifted sensor.**
- **Analysis techniques to predict drifts that indicate sensor measurements outside their operating range.** These techniques include the characterization of empirical drift-estimation error based on available data and the generalization error that applied to future test dataset using methods such as Vapnik-Chervonenkis (VC) theory [21].
- **UQ techniques that can be integrated into drift detection and estimation, as well as online recalibration, virtual sensing, and predictive monitoring.**

The US Department of Energy (DOE) Office of Nuclear Energy (NE) sponsored research into some of these questions through other programs. Ongoing research is addressing the development of virtual sensors for short-term replacement of data from a failing sensor while that sensor is repaired or recalibrated. Other research is evaluating approaches for calibration-interval extension incorporating OLM. This study focuses on new algorithms based on advanced data analytics that may be capable of providing improved sensor-response estimates and confidence bounds in the sensor estimates for unknown future measurements.

This report focuses on detecting and quantifying sensor drifts caused by calibration changes and conditions such as sensing line blockage or leaks (some failure modes of pressure transmitters) to a limited extent. The primary objective in addressing this question is to generate UQ estimates that are improved by considering the information specific to the underlying NPP systems. Using the data available under normal or known drift conditions, models can be trained to detect and estimate sensor drifts. The training error is computed and used as a component of generalization error, which depends on the properties and parameters of the method. The estimates of drift and the associated generalization error bounds could be used as correction terms to measurements and as input to recalibration processes, although this is the subject of ongoing research.

Note that automatically distinguishing between sensor drift and process change is also a challenge. This is not addressed in this research. Instead, it is assumed that the operator is knowledgeable about expected changes in the process.

3. ALGORITHMS FOR ONLINE SENSOR DRIFT DETECTION AND ESTIMATION

3.1 Data Analytics Methods: Overview

The goal of OLM methods is to enhance the reliability of NPP measurements through improved accuracy and increased reliability of the sensors used to monitor key parameters. The literature[22,23] identifies several classes of algorithms that may be applied to OLM. All these techniques rely on comparison of measurements against data from baseline conditions (normal or faulted) for drift detection. Some approaches use these data directly, while others use features or signatures computed from data. However, other OLM techniques use a model to predict the correct plant measurements, with deviations from these predictions in the measured data indicating calibration drift. The algorithms themselves may rely on data from a single sensor or a redundant sensor set, or they might use measurements from different sensor types such as a thermocouple, a resistance temperature detector, or pressure transmitters.

While physics-based models may be applied to predict the correct plant parameters, challenges associated with identifying the model parameters themselves have led to the application of models that are induced from historical data. Such data analytics methods use historical data to learn the model parameters and then apply these models to predict measurements.

The focus of this effort is on such model-based OLM methods in which the models are derived using one or more data analytics approaches that rely on historical measurements from sensors that operate normally and/or that are failing or drifting. Specifically, these approaches estimate the output of a sensor by fusing measurements from other sensors and using the difference between a measurement and its estimate as a measure of drift or error. This approach is particularly effective for estimating large drift, because the drifting sensor is not used in the estimate. A special case consists of redundant sensors that measure the same underlying physical quantity, which in principle provide "close" measurements.

Data analytics methods for OLM must:

- Enable development of robust models with accurate prediction of plant measurements. While specific tolerances on the prediction accuracy are not available, generally, prediction accuracy in excess of 90% is likely to be necessary to limit false or missed detection.
- Access historical data. Typically, the volume of data needed for model development will depend on the specific algorithm. For drift detection under steady-state operating conditions, the historical data must largely come from normal operations. For fault detection and diagnostics, especially if the intent is to use methods based on pattern recognition, the data will also need to include examples of likely fault conditions.
- Provide estimates of the uncertainty associated with the model prediction and, if possible, the measurement. The level of uncertainty will be a key parameter in setpoint analyses for sensor calibration-interval extension. As with accuracy, apparently there is no specific guidance on the uncertainty levels or on the confidence bounds. This research targets generalization equations based on Vapnik's theory that bound the prediction error at a specified confidence level for predictions.

3.2 Candidate Data Analytics Methods

Of the large number of potential methods applicable to OLM tasks, this study focuses on a subset of auto-associative and hetero-associative models. In general, auto-associative models support the processes of pattern recognition and completion, whereas hetero-associative models support the processes of paired-associate learning and sequence generation.[24] Auto-associative models use the same data as input and output, and they typically maintain fewer parameters than more general models. The models work to predict the entire input/output sequence from a smaller sample of the input sequence. For example, given the sentence fragment "That's one small step for a man, a giant _____," those familiar with Neil

Armstrong and US space exploration will complete the quote "That's one small step for a man, a giant leap for mankind." In this example, the incomplete sentence or subset of the input is sufficient to recreate the entire quote or output.

In the case of OLM, AAKR uses measurements from a group of sensors are used to predict the responses from all these sensors, assuming normal sensor operational conditions. Given the need to predict the measured data from actual measurements, auto-associative models generally require highly correlated variables. Under normal sensor operational conditions, the prediction error can be low if the model parameters are properly tuned. Under conditions in which one or more measurements drift, the prediction error increases and may be used to detect the presence of a sensor drift. However, the inclusion of drifted signals as part of input results in a higher prediction error for the drifted sensor and other sensor measurements. Therefore, the performance of auto-associative models is often assessed by auto-sensitivity and cross-sensitivity metrics which indicate the robustness of the models to the drift or fault. Hetero-associative memory recall an associated piece of datum from *one* category upon presentation of data from *another* category and can be viewed as a generalized version of the auto-associative memory. We present information fusion methods that use non-drifted sensors to reconstruct the correct sensor signal.

In both auto- and hetero-associative model-based approaches to OLM, a fundamental aspect is the use of information from multiple sensors in the model-prediction process. A key assumption is that information-fusion methods in this context use the structure and other information from physical laws of parameters of the underlying system for drift and sensor-error estimation.

3.2.1 Auto-Associative Algorithms for Sensor-health Evaluation

The AAKR method is considered in this study mainly for its potential in using measurement data to infer model parameters, its extensive historical use in OLM in the nuclear industry, and its ability to provide prediction error bounds (generalization error bounds). AAKR [25] is a nonparametric, memory-based modeling technique that takes an input historic-memory vector and a query vector to calculate the predicted measurement vector. This method is widely known for its ease of use and its ability to attain a high level of prediction accuracy. AAKR relies on statistical relationships between variables to make predictions about their true values. For this reason, inputs to the kernel regression model must be well correlated to one another. Sensors are grouped based on their linear intercorrelation. Once the groups are defined, an AAKR model is trained for each group and used to reconstruct signals of the query vector as expected under normal conditions.

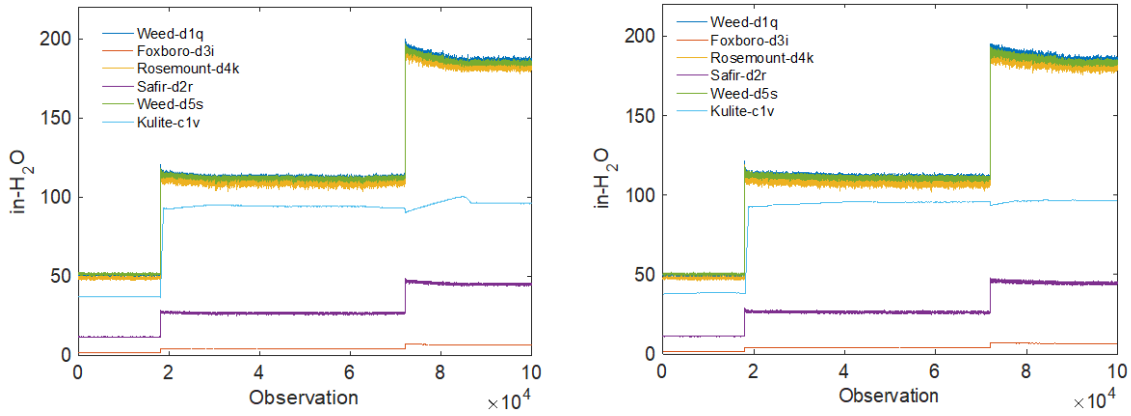
3.2.2 Information Fusion for Drift Detection and Estimation

The information fusion method, which is based on multiple sensors for drift estimation, was developed under this project. The sensors of a power plant measure variables that are typically related to each other through the underlying system physics. Such relationships provide regressions with smooth or non-smooth but bounded variation properties that are conducive to machine learning analytics [26]. Informally, the *bounded variation* in this case means that discontinuities or non-smooth sections have a finite number of finite differences. To be considered conducive, in closed systems such as a primary coolant loop, critical trends must be discernable above the sensor noise. For example, in laboratory flow loop scenarios that provided the data sets used here (see Section 4.1 for details), the source temperature is increased twice during the experiment, starting from ambient temperature. At time $t = 900\ s$, it is increased from ambient temperature to $70°F$ and at time $t = 3{,}600$ seconds, it is increased to 90°F. The sensor measurements collected at 20 Hz show a corresponding response in two-step increases at $x = 18{,}000$ and $x = 72{,}000$ (Figure 1) as a result of higher flow rates acting to maintain the loop control temperature setpoints. These sensors have different measurement units and ranges:

i. Differential pressure sensors:
    a. Weed-d1q: 0–850 in. H_2O
    b. Foxboro-d3i: 0–50 psi

c. Rosemount-d4k: 0–750 in. H_2O
d. Safir-d2r: 0–259 kPa
e. Weed-d5s: 0–250 in. H_2O

ii. Pressure sensor: Kulite-c1v: 0–100 psi



(i) no-drift training dataset: Scenario 10 (ii) drifted Kulite-c1v test dataset: Scenario 14

Figure 1. Training and test measurements for sensor fusion method for estimation of drift.

These types of relationships can be exploited to learn a regression fusion function for measurements of a chosen sensor as a function of measurements from other sensors. This data can then be used to estimate measurements that are "expected" under no error conditions. When a sensor drifts (due to calibration shift or sensing line blockage), the difference between its actual measurements and predicted regression values provide estimates of the sensor drift that can assist in identifying faulty sensors.

SVM and EOT methods are chosen to estimate the regression function to capture two different properties: smooth and non-smooth functions, respectively. The SVM method is based on nonlinearly transforming the feature space X so that it is suitably expanded into regression regions based on Y. The result a smooth regression function f_{SVM} from a class F_{SVM} consisting of smooth functions resulting from Gaussian kernels. The EOT method is based on boosting a collection of classification trees that are customized to fit the training data using the AdaBoost method. The resulting highly non-smooth regression function f_{EOT} is from a function class F_{EOT} consisting of a large collection of decision trees. These algorithms are described in more detail in Section 4.2.3 and in the literature [3,27,28,29,30,31,32].

3.2.2.1 ***Support vector machines.*** SVMs are kernel-based machine learning tools used for classification and regression [21]. The SVM concept is based on constructing hyperplanes to linearize a classification or regression task and developing support vectors that minimize the error margin. In SVM regression, the goal is to find the function f_{SVM} for the input data that has at most ε deviation from the output data while being as flat as possible. SVM regression tries to approximate all data pairs within a 2ε band using f_{SVM} while allowing some slack points beyond the 2ε band. A constant determines the tradeoff between flatness of f_{SVM} and toleration of points beyond ε. The flatness of the f_{SVM} indicates that the weight vector is minimized, and when slack points are included, then the minimization includes the weight vector and additional error introduced by the slack points. For nonlinear input data, the input vector is mapped into the high-dimensional feature space through some nonlinear mapping to make it possible to perform linear regression.

3.2.2.2 ***Ensemble of trees***. Regression tree ensemble is a predictive model composed of a weighted combination of multiple regression trees. In regression problems, boosting builds a series of regression trees in a stepwise fashion and then selects the optimal tree using an arbitrary differentiable

loss function. *Boosting* means that each tree is dependent on prior trees. The algorithm learns by fitting the residual of the preceding trees. Thus, the prediction model is an ensemble of weaker prediction models. Boosting in a decision tree ensemble tends to improve accuracy with some small risk of reduced coverage.

3.2.2.3 ***Nonlinear principal component analysis*** . Principal component analysis (PCA) is a popular statistical method for extracting information from measured data. PCA finds axes of significant variability in the data by forming linear combinations of variables. Nonlinear principal component analysis (NLPCA) is an extension of linear PCA. While PCA identifies linear relationships between process variables, NLPCA extracts both linear and nonlinear relationships. NLPCA can be represented by two submodels: the mapping model and the demapping model.

Given a data matrix *X*, the mapping model computes the nonlinear principal component *T*, and the demapping model computes estimates of the data. NLPCA is used to model normal process behavior, and faults are then detected by checking the observed behavior against this model. The proposed work [33] used a five-layer neural network NLPCA model.

3.2.2.4 ***Principal component regression.*** PCA uses an orthogonal transformation to convert a set of possibly correlated variables into a set of linearly uncorrelated principal components (PCs). PCA provides a way to compress data by reducing the dimensions of predictor variables without much loss of information. The original *n* correlated sensors are projected to a new space, with the *n* uncorrelated variables in the order of the amount of the variance. The PCs are the eigenvectors of the covariance matrix of the input matrix, which are proportional to the variance amount.

Principal component regression (PCR) [34] is a regression method in which the PCs are used as inputs instead of original sensors to predict an output sensor. The generated PCs meet the assumption of linear regression: that the inputs are uncorrelated. Because most sensors are correlated, PCR provides a much smaller input space based on PCs, which reduces the computing cost significantly.

3.2.2.5 ***Other regression methods***. There are several different types of regressions, each with its own strengths and weaknesses. In addition to the methods discussed above, a set of common regression methods was evaluated for reference. Those methods include linear regression, k nearest neighbor regression, decision tree regression, gradient boosting regression, random forest regression, and ridge regression.

Linear Regression [35] models the relationship between a single input independent variable and an output-dependent variable and is completely made of linear variables. In the more general case, multivariable linear regression captures the relationship between multiple independent input variables and an output-dependent variable. Linear regression assumes normality in variables and fails in the case with high collinearity among the feature variables.

Ridge Regression [36] makes the same assumptions as linear regression, except it does not assume normality. In addition, it adds a small squared bias factor to the variables, pulling the variable coefficients away from their rigidness and greatly reducing their variance. *Decision tree regression* is a nonparametric supervised leaming method that learns from data to approximate a sine curve with a set of if-then-else decision rules. The fitness of the model depends on the complexity of the decision tree. *k nearest neighbor regression* uses a weighted average of the *k* nearest neighbors weighted by the inverse of their distance for estimating continuous variables. Choosing the optimal *k* value is important since the right *k* value reduces the overall noise without overfitting or underfitting for the data.

Gradient Boosting Regression [37] is an additive ensemble model that consists of three components: a loss function to be optimized, a weak learner to make predictions, and an additive model to add weak learners to minimize the loss function.

Random Forest Regression [38] is an ensemble machine learning technique that uses multiple decision trees and a statistical technique called *bagging* to reduce high variance and bias. Instead of merely averaging the decision trees, it uses random forest of training observations when building trees, and it uses random subsets of features when splitting nodes. Combining several decision trees into a single model brings the predictions closers to the true value on average.

All the techniques listed above may be used in either an auto-associative or hetero-associative mode. In this study, these techniques are used in a hetero-associative mode, with the goal of predicting the output of one or more sensors using information in measurements from other sensors.

3.2.3 Performance Evaluation Metrics

Modeling methods for OLM should produce accurate, repeatable, robust results and should provide an estimate for uncertainty of the predictions. According to NUREG/CR-6895 [39], the performance of auto-associative OLM systems is measured in terms of three metrics: accuracy, auto-sensitivity, and cross-sensitivity. This work includes three more metrics—time to detect drift onset, detection rate, and false alarm rate—to provide an objective set of metrics for evaluating the performance of these algorithms. Based on the differences between auto-associative and hetero-associative models, not all metrics will be applicable, and it is unlikely that any single method will outperform in all metrics, as optimizing one metric typically degrades another.

1. ***Accuracy*:** Accuracy measures the ability of a method to correctly model and accurately predict the sensor measurements. It is characterized by the mean squared error (MSE) between the predictions and measurements. The MSE (A), corresponding to a sensor is given by

$$A = \frac{1}{N}\sum_{i=1}^{N}(\hat{x}_i - x_i)^2 \quad (1)$$

   where x_i and $\hat{x}_i$ are the measurement and its prediction, respectively, for observation *i*, and *N* is the number of observations. For a drifted measurement, the MSE is expected to be high as the model predictions are generally of the measurement under non-drifted conditions. The root mean square (RMS) error given by the square root of the MSE is also used as an estimate of accuracy in some cases.

2. ***Sensitivity*:** This is a measure of the robustness of the models. A robust model is defined as one that produces small changes in its output in response to small errors in its inputs. Model sensitivity is generally defined as a measure of the change in the prediction of the i^{th} variable $(\hat{x}_i)$ produced by a change in its respective input (x_i):

$$S_i = \frac{\Delta\hat{x}_i}{\Delta \mathrm{x_i}} \quad (1)$$

   a. ***Auto-sensitivity*:** Auto-sensitivity is a measure of a model's ability to correctly predict a faulty sensor that is included in the input of the model [39]. Auto-sensitivity indicates the degree to which the faulty sensor prediction will be impacted by itself. It takes as input a sensor *i*'s prediction with no fault in the input and with a faulted input, $\hat{x}_i$ and $\hat{x}_i^{drift}$ respectively, and sensor *i*'s unfaulted-input value and drifted-input value, x_i and x_i^{drift}. For experimental analysis, to calculate auto-sensitivity, artificial drift is introduced by adding a standard deviation to the original signal.

   b. ***Cross-sensitivity*:** This value measures the effect a drifted or faulty sensor input has on the predictions of other sensors [39]. In other words, this value measures the effect that a faulty sensor input *i* has on the predictions of sensor *j*.

3. ***Time to detect drift onset.*** This value is defined as the observation point at which the method detected drift in the measurement after the onset of drift.
4. ***Detection rate.*** This value is measured by the fraction of time drift that is declared by a method during the period the sensor is drifted.
5. ***False-alarm rate.*** This is defined as the fraction of time that a method incorrectly identifies measurement drift during the time that the sensor is operating normally.

In addition to the above metrics, several other measures of performance may be identified, such as scalability and computational cost. *Scalability* is a measure of the ease of scaling of the method from a few (tens) to many (hundreds to thousands) sensors and is often measured indirectly through the *computational cost* (time, computing cycles, or memory needed to generate a given number of predictions), which may scale nonlinearly with the number of sensors.

4. PRELIMINARY RESULTS

This section includes the initial results of AAKR, PCR and information fusion methods in terms of their performance on testbed datasets and generalization error equations.

4.1 Datasets

To investigate data analytics methods for drift detection and estimation for NPPs, two sets of measurements were used. The first set was collected using a laboratory-scale flow loop located at Analysis and Measurement Services (AMS Corporation, Knoxville, Tennessee) while the second was collected during a fuel-irradiation test program conducted in the Advanced Test Reactor at Idaho National Laboratory.

A schematic of the laboratory-scale flow loop is shown in Figure 2. The loop includes a heater, a primary loop pump, and a heat exchanger. A chiller is connected to the secondary side of the heat exchanger. Figure 2 also indicates the installation location of twelve pressure transmitters, six resistance temperature detectors, and an electromagnetic flow meter. The installation of these sensors emulates the two main characteristics of sensors in NPPs: redundancy for measuring key parameters and correlated data from the sensors.

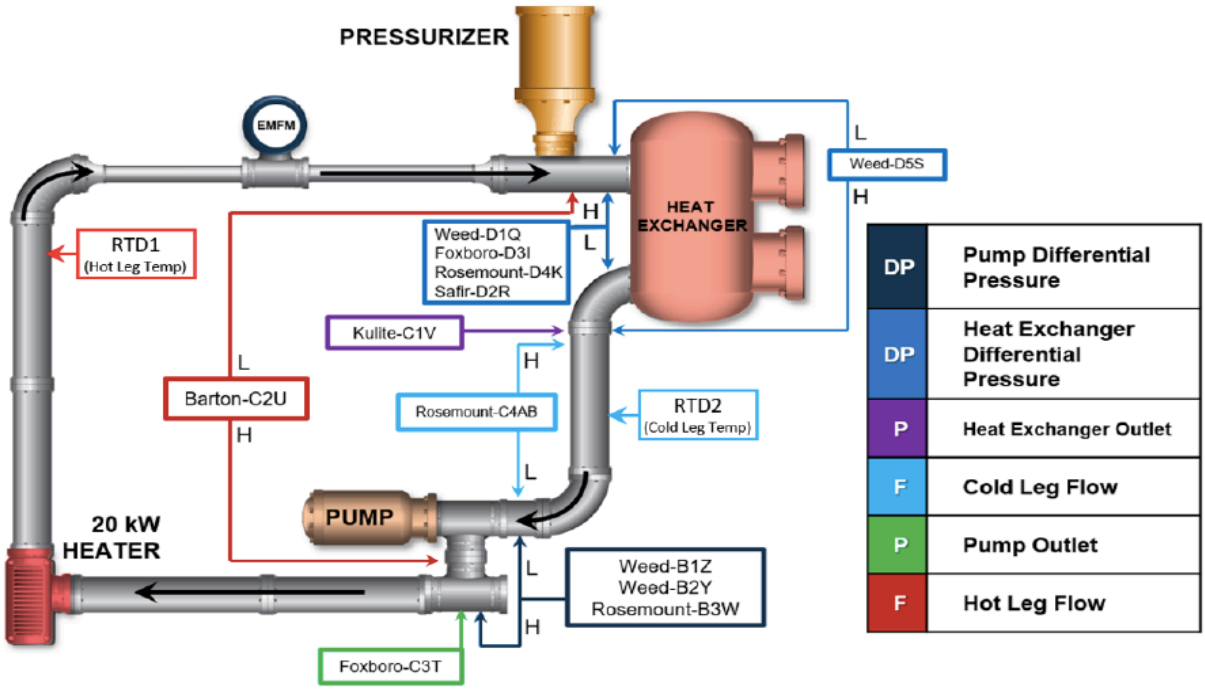


Figure 2. AMS test loop layout [40].

Twenty scenarios were used to collect measurement datasets using this flow loop [40]:

- 2 under normal operation (Scenarios 1 and 10: labeled as AMS3 and AMS12);
- 8 simulated calibration changes (Scenarios 2-9: labeled as AMS4-AMS11),
- 4 simulating blockages (Scenarios 11-14: labeled as AMS13-AMS16),
- 3 simulating minor leaks (Scenarios 15-17: labeled as AMS17-AMS19),
- 2 simulating air voids (Scenarios 18-19: labeled as AMS20-AMS21), and
- 1 simulating electromagnetic interference (Scenario 20: labeled as AMS22).

All twenty scenarios of the AMS dataset were collected with the loop operating over three operational ranges: low, in which the heater was OFF and the chiller was ON; medium, in which the heater and the chiller were both ON; and high, in which the heater was ON while the chiller was OFF. For Scenarios 2–9, in which simulated calibration changes were introduced, the simulated drift was initiated in a Rosemount-d4k differential pressure (DP) sensor, which was one of a redundant set of four DP sensors monitoring the pressure drop across the heat exchanger: the pressure difference between hot and cold legs. While redundant, these sensors are not identical, and they are complementary in that they have different units or ranges. Weed-d1q, Weed-d5s, and Rosemount-d4k measure in units of inches in H_2O but have different ranges: namely, 0–850, 0–250, and 0–750, respectively. Foxboro-d3i measures in PSI units in a range of 0–50, and Safir-d2r measures in units of kPa in a range of 0–259. For other scenarios, faults were introduced to affect again Rosemount-d4k (Scenarios 11-13 and 15-19), Kulite-c1v (Scenario 14) and Barton-c2u (Scenario 20). In all scenarios, measurements were collected at 20 Hz for a period of approximately 85 minutes.

While these redundant sensors can be identified visually from the layout in Figure 2, a correlation analysis between sensors, as shown in Figure 3, also confirms information about the strongest and the weakest correlated sensor pairs. This may be useful, as the analyst may not have access to full design drawings and the layout of the sensors in a flow loop.

The correlation coefficients from the AMS dataset summarized in Figure 3 show a strong correlation between the various DP sensors, as well as among the pressure transmitters and the groups of temperature sensors. Cross correlations between the different sensor types are also high, with a few exceptions (such as between the primary loop RTDs and the pressure transmitters). Additional analysis is needed to determine whether the data are correlated with a time lag or if there are other causes for the lack of correlation between these sensors.

Given that the Rosemount-d4k is the drifted sensor, sensors are used that are strongly correlated with Rosemount-d4k, such as Weed-d1q, Foxboro-d3i, and Safir-d2r, as shown in Figure 3, to identify the drift in the signal pairs. The obvious advantage of this approach is to identify redundancy in measurements and to account for the redundancy in detection and prediction calculations.

The second data set, which is referred to as the AGR dataset in this report, was a subset of data collected during the Advanced Gas Reactor (AGR) fuel-irradiation test program conducted in the Advanced Test Reactor at Idaho National Laboratory. This subset of the AGR dataset contains thermocouple (TC) measurements of graphite in six capsules. Each capsule underwent 12 irradiation cycles from February 2007 to July 2009.

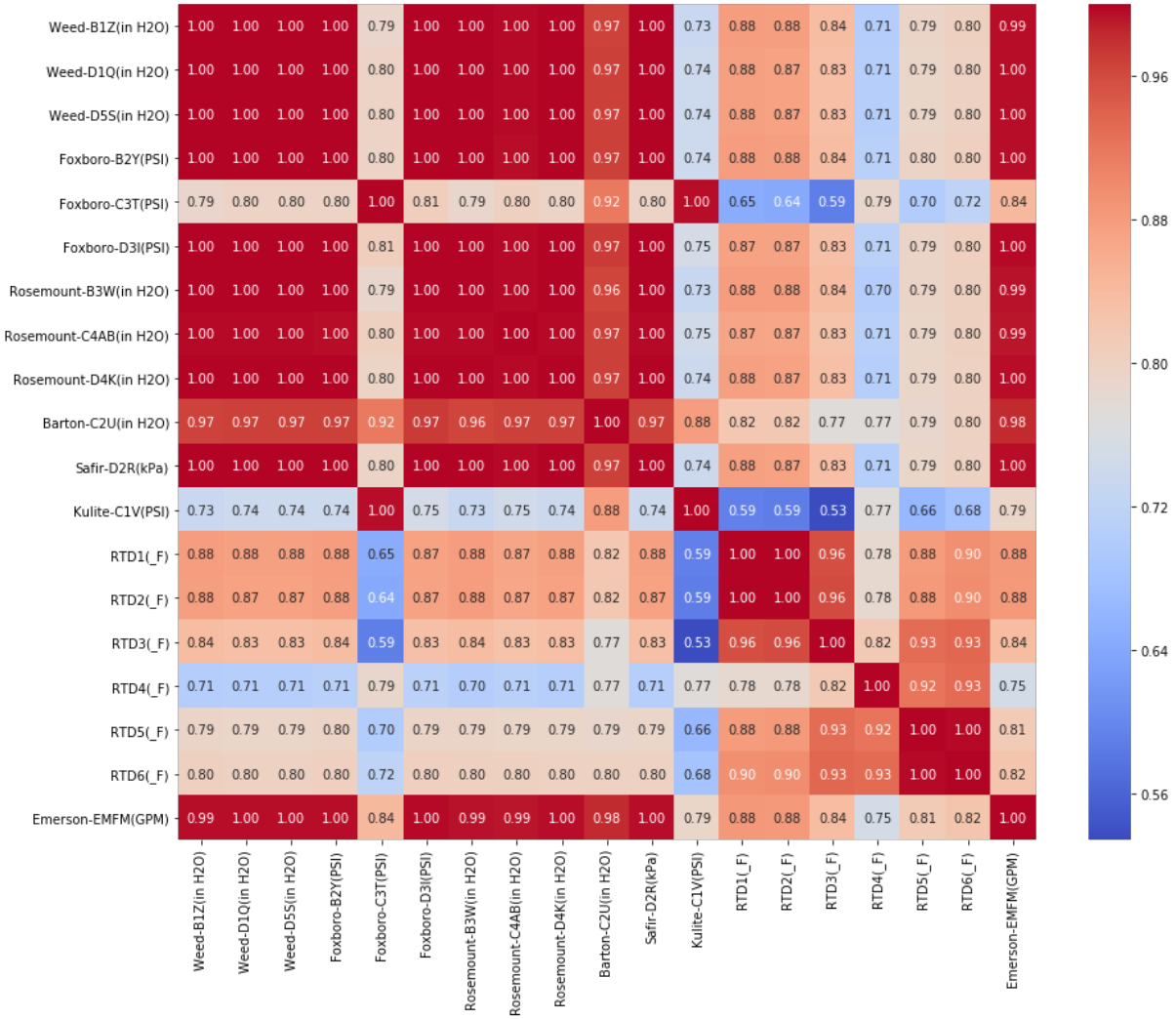


Figure 3. Correlation between AMS test loop sensors.

The AGR dataset contains measurements from five TCs in the experiment. Due to failures of the TCs, only Capsule 6 has more than three TCs surviving with data recorded and was used in the analysis. The analysis focused on the detection of drift and the detection of failures of TCs. Initial results indicated the potential for detecting the onset of drift in the TC data using some of the algorithms described earlier; however, the information available did not include true state information (i.e., independent information on when the TCs began drifting or failed completely). As a result, the analysis of this dataset is ongoing and is not reported in this section. Appendix B contains a summary of the analysis to date with additional analysis ongoing and planned for reporting in subsequent milestone reports.

4.2 Initial Results

This section begins with the AAKR method and establishes it as a baseline and then compares the performance with other methods investigated in this research based on the metrics discussed in Section 3.2.3. To facilitate readability and establish consistency, the following definitions are provided:

False alarm: when the model erroneously indicates an error when there is none.

Missed alarm: a false negative where the model shows no indication of error despite an error being present.

In all methods investigated in this research, the focus to date has been largely on measurement prediction. The immediate application is the ability to estimate drift, i.e., demonstrate that the model accuracy is high in cases with no measurement drift and that the MSE may be useful as an input for detecting the onset of drift. Rudimentary drift detection techniques using pre-determined and likely sub-optimal thresholds have been applied in some instances, where threshold selection required knowledge of the onset of drift in the data. Methods and procedures are being studied for relaxing this requirement and performing drift detection without access to this information, and will be reported in follow-on reports.

4.2.1 Drift Detection and Estimation Using AAKR

Eight scenarios from the AMS dataset are considered (runs AMS4 through AMS11) with drifted sensors to predict drift using the AAKR model. Rosemount-d4k is the drifted sensor in each case. AAKR is evaluated using two sets of training data: the first set of inputs consists of measurements from all sensors with the goal of predicting all of them, and the second set consists of a highly correlated subset of sensors. Figure 3 depicts the correlation between different sensors in the AMS loop. In auto-associative models, the outputs mimic the inputs. The input to an AAKR consists of drifted and non-drifted sensors, and the output is ideally corrected sensor signal.

The AMS loop consists of 102,400 observations per sensor. Noting the computational cost for training AAKR on the entire set of observations, a smaller subset of 25,600 memory vectors—the first 1/4th of the total observations—is also considered as training data. The average accuracy (as defined in Section 3.2.3) on the training data was 1.87×10^{-4}. While this is considered acceptable accuracy, the computational effort in terms of memory and computational cost was significant.

To overcome the limitation of computational resources in training the AAKR model, a representative sample of 500, 5,000, 12,800, and 25,600 memory vectors is chosen using the minmax vector ordering method presented in NUREG/CR-6895 [39]. The empirical results showed that using 5,000 memory vectors presents better accuracy with relatively low computing cost. This subset selection significantly reduced the computational need for training AAKR, with only a slight reduction in accuracy at 4.9×10^{-4}. As the computation time is typically dependent on the underlying hardware, execution time is not reported, but it will be included in future reports once an objective measurement is determined.

Given these encouraging results from memory vector subset selection, 5,000 memory vectors are used as the training vector length for the AAKR model in subsequent analyses. Other metrics discussed in Section 3.2.3, such as sensitivity, were also computed for this model; analysis of these metrics is ongoing.

AKR performs best when the process variables present a strong collinearity among themselves. Having a choice of all 19 sensors with no strong collinearity reduces the capability of AAKR to satisfy the missed alarm and false-alarm rates. Therefore, a set of nine pressure or pressure differential sensors were chosen that are highly correlated with the drifted Rosemount-d4k sensor. Sensors having a low correlation with Rosemount-d4k sensor or environmental sensors that remain constant throughout the operation were excluded. Different combinations of highly correlated sensors were tested as a training data to the AAKR. One of the tests included using the training data without the drifted sensor (drift-free dataset) to observe whether the prediction performance would improve. A model was chosen that presented the lowest root mean square error (RMSE) and high accuracy for drift estimation. The accuracy was computed by comparing the estimated and real drift, which is the difference between the drifted sensor and the redundant drift-free sensor.

AAKR performs best when the process variables present a strong collinearity amongst themselves. A choice of all 19 sensors, with no strong collinearity, reduces the capability of AAKR to satisfy the missed alarm and false-alarm rates. Therefore, we choose a set of nine pressure or pressure differential sensors that are highly correlated with the drifted Rosemount-d4k sensor. We excluded those sensors that possess a low correlation with Rosemount-d4k sensor or environmental sensors that remain constant throughout the operation. We tested different combinations of highly correlated sensors as training data to the AAKR. One of the tests included using training data without the drifted data (drift-free dataset) to gauge the effectiveness of the model in correcting the signals. We picked the model that presented the lowest root mean square error (RMSE) and high accuracy for drift estimation. We computed the accuracy by comparing the estimated and real drift, which is the difference between the drifted sensor and the redundant drift-free sensor.

Figure 4 shows the residual between the measured and predicted sensor data for the Rosemount-d4k DP sensor in different AMS dataset scenarios. We perform drift detection using a carefully chosen threshold. If not done methodically, such a detection approach, as shown in Figure 4, may lead to an increase in the false negatives. Therefore, the false alarm and missed alarms rates depend on the selected threshold value. Methods for optimal selection of the threshold, for instance, using techniques such as the Neyman-Pearson lemma, will be necessary when applied to data where the onset of drift is not known *a priori*. The dataset AMS4 results in a large residual independent of the algorithm used and we are still investigating reasons for this. AMS4 through AMS11 are all simulated calibration change experiments (see Section 4.1 for dataset descriptions).

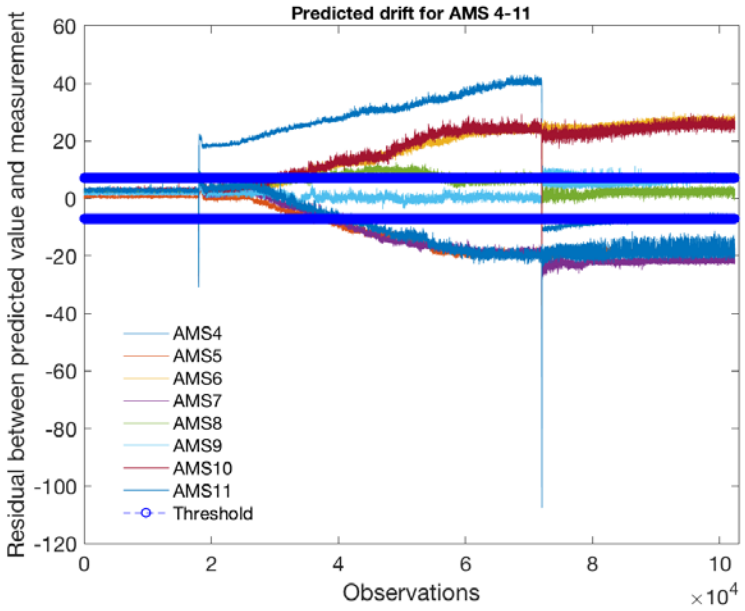


Figure 4. Predicted drift for differential pressure sensor Rosemount-d4k.

Input data are normalized before the model is trained to avoid sensors with large values taking higher importance. Figure 5 shows the residuals of the nine sensors (only one sensor is drifted) in AMS5 with normalization. Comparing Figure 4 and Figure 5, it is evident that the residual is much noisier when data are not normalized. This presents a challenge to establish an effective threshold with relatively low missed and false alarms. The blue line is the residual of a fault-free sensor, while the red line is the residual of a drifted sensor. To balance the missed and false alarms, the threshold selection was in a manner for late detection of drift. It may be effective to use denoising or smoothing algorithms to reduce excessive noise. However, these algorithms were not used in the predictions in this research and may be included in the future.

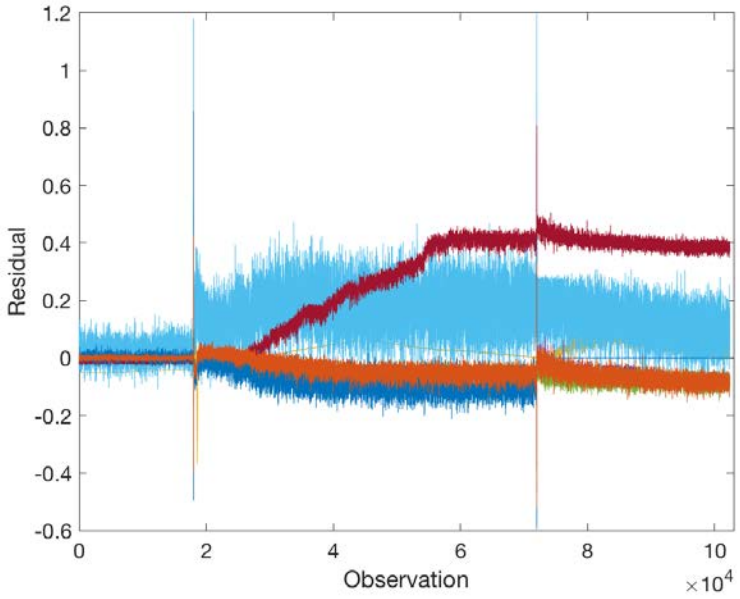


Figure 5. Residuals of nine sensors in AMS5 with normalization.

Residual is defined as the difference between predicted value and the ground truth. Figure 6 shows the residual between the drifted Rosemount-d4k sensor and the redundant ground truth, Weed-d1q sensor. It also depicts the drift predicted in other Scenarios 7–9 (AMS 9–11).

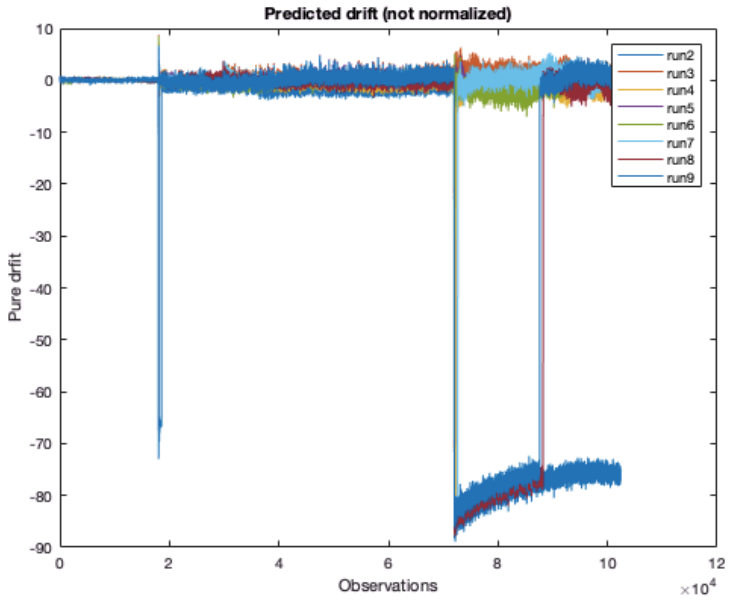


Figure 6. Difference between predicted drift and ground drift.

The residual and fault hypotheses of AMS5 using the AAKR model can be seen in Figure 7, which demonstrates the detection capabilities. The thresholds were selected to predict the first observed drift based on the known start times for drift (thresholds are shown as red dotted lines in Figure 7).

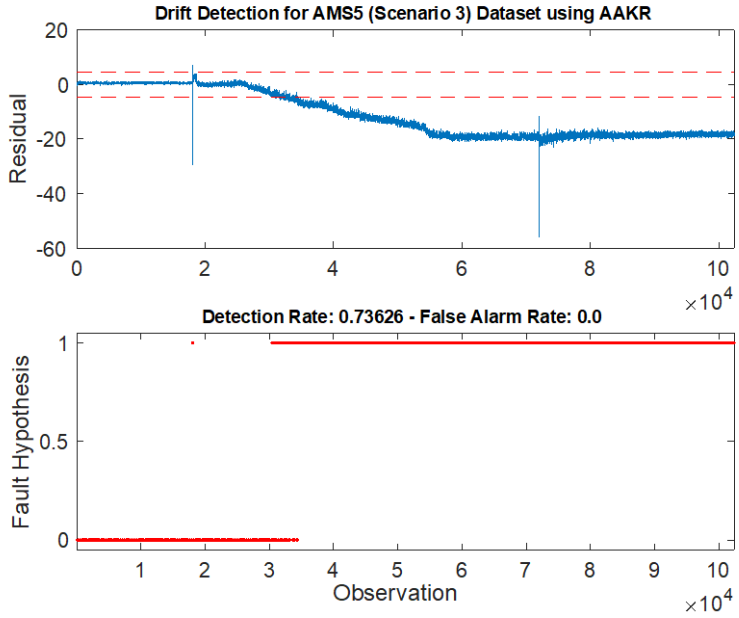


Figure 7. The residual and fault hypothesis of AMS5.

4.2.2 Principal Component Regression

The advantage of using PCR over AAKR is its low computing cost and lower cross-sensitivity. This means that if a drifted sensor is included in the generation of the principal components, it does not significantly impact the regression results. However, the principal components are highly affected by peaks in a signal, so a peak smoothing filter was applied at the points in the signal with parameter changes.

As with the AAKR analysis, the PCR algorithm was evaluated using eight scenarios from the AMS dataset (runs AMS4 through AMS11). All the pressure sensor data are used except for the drifted Rosemount-d4k sensor to generate the principal components. The data are normalized, and after generating the principal components, it can be observed that choosing two PCs allows us for capture of more than 99.9% of the information in the drift-free data. Linear regression is then employed to predict the Rosemount-d4k sensor using the principal components. The linear regression result is expressed as follows:

$$p_{D4k} = 0.3858v_1 - 0.1408v_2 \quad (2)$$

where p_{D4k} is the pressure reading of the Rosemount-d4k transducer, and v_1 and v_2 are the two principal components.

The RMSE of the drift-free test data is 0.0126, and the drifted Rosemount-d4k values of the AMS4-11 are predicted using linear regression. The residual is computed as the difference between the predicted value and the measured values. Figure 8 shows the predicted drift for scenarios AMS4-AMS11 using PCR. Comparing Figure 4 and Figure 8, it can be seen that AMS4 has a large residual using both AAKR and PCR. This leads to either false alarms or large thresholds that result in missed alarms.

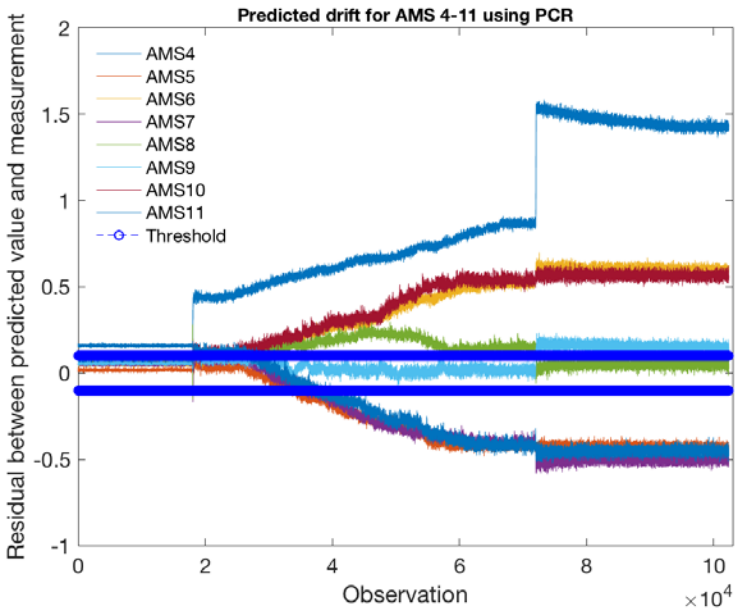


Figure 8. Predicted drift with PCR exclude the drifted sensor.

As an alternative to seeing the effect of the drifted sensor in training the PCR model, the Rosemount-d4k drifted sensor was used in the training data. Figure 9 depicts that PCR behaves robustly against the drifted sensor, and the predicted drift is still consistent with the model trained with drift-free data.

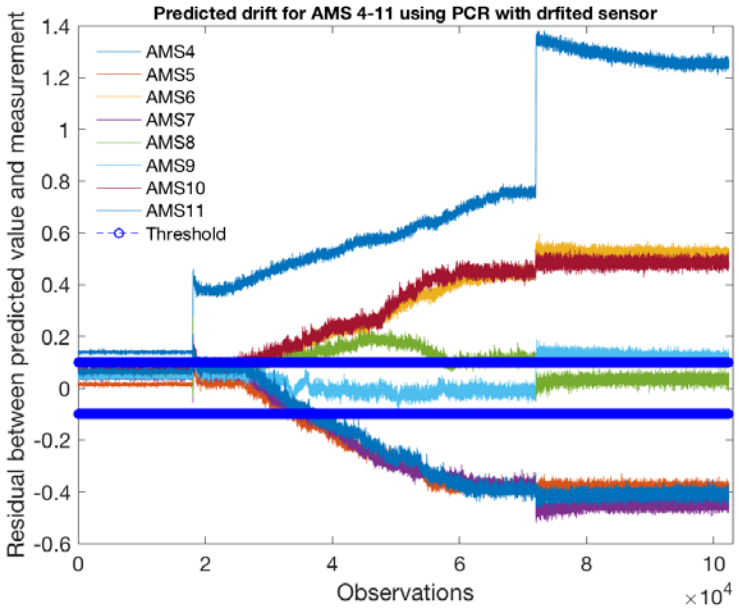


Figure 9. Predicted drift with PCR include the drifted sensor.

The thresholds are carefully defined for detecting the first observed drift considering the false and missing alarms. When thresholds are small, they are sensitive to small drift, which could give early detection of drift. However, small thresholds also result in more false alarms because the random noise in the residual could intersect with the defined thresholds. From intuition, while early drift detection is preferred, it is not essential, because a drifted sensor in its early drift stage does not noticeably impact the operation. In contrast, false alarms could be harmful, as they can disturb the operation. Therefore, thresholds are selected to tradeoff between false alarms and missed alarms. Ideally, the threshold should be large enough to reduce false alarms, but this also results in a relatively large number of missed alarms in some datasets. Figure 10 presents the detection results, the detection rate in the mid-range, and the false-alarm rate in the low-range of the PCR for scenarios AMS 4–11. The mid-range starts at observation 18,000, and the high range starts at 72,000.

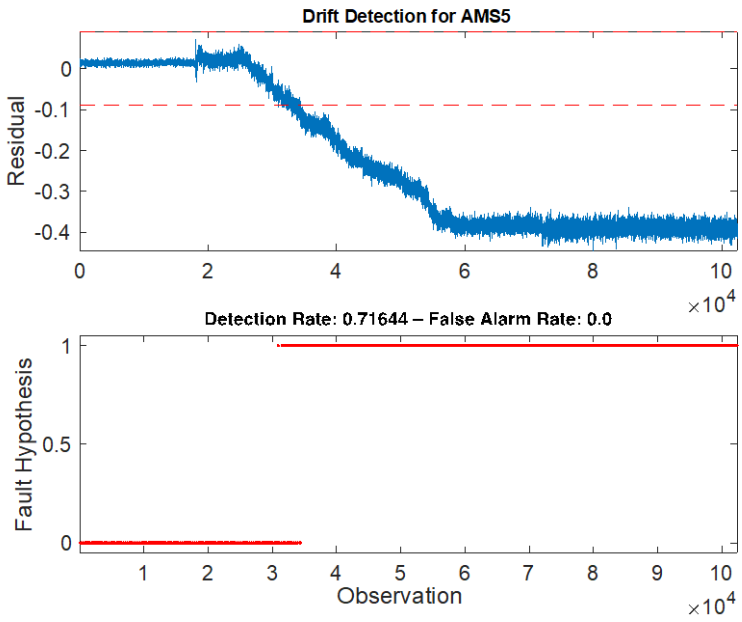


Figure 10. Detection results for AMS5.

Table 1 presents a comparison of performance metrics, which includes three key performance metrics for AAKR and PCR: the time at which the first drift was detected, detection rate, and false-alarm rate for various datasets.

Table 1. Performance matrix of AAKR and PCR.

Model/Algorithm	AMS Dataset Number	First Observation Detected (sample number)	Detection Rate (%)	False-alarm Rate (%)
AAKR	Fault-free test data	0	1[1]	0[2]
	AMS4	18000	99.981	1
	AMS5	18018	73.626	0
	AMS6	18025	78.874	0
	AMS7	18035	68.094	0
	AMS8	18020	84.798	0
	AMS9	18048	6.0185	0
	AMS10	18030	86.022	0.12778
	AMS11	18021	69.515	0.20557
PCR	Fault-free test data	0	1	0
	AMS4	18000	99.993	1
	AMS5	30896	71.644	0
	AMS6	18041	80.183	0
	AMS7	18072	67.719	0
	AMS8	18030	85.048	0
	AMS9	18269	0.1185	0
	AMS10	18040	91.894	0.04445
	AMS11	18030	70.754	0.00556

NOTE: *First observation detected: the index of the first true positive alarm. In an ideal situation, the number for this application should be 18000.*

1 Detection rate: the fraction of the detected drift/all the drift in the medium range because medium range is the range to which the drift is added. In an ideal situation, this number for this application would be 1.

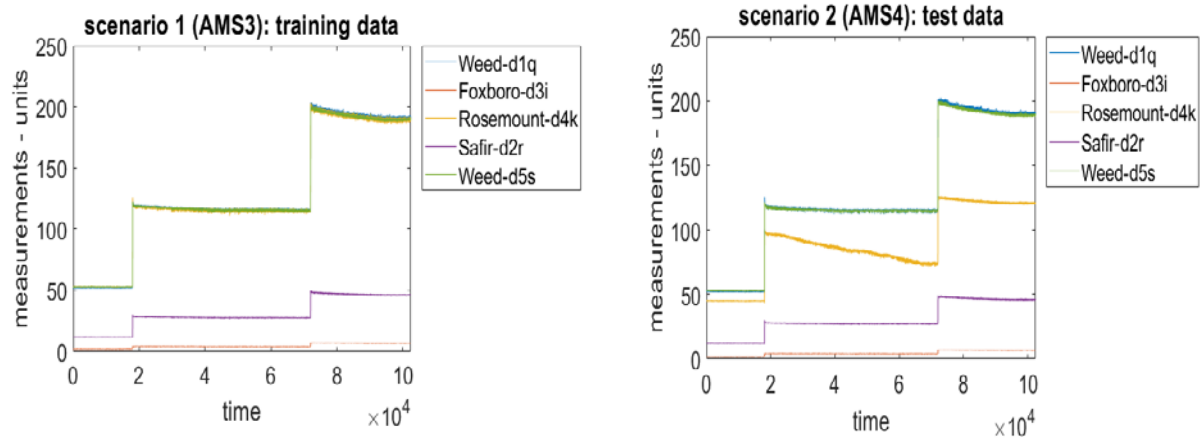
2 False-alarm rate: the fraction of the false alarms is in the low range because there is no drift in the low range. In an ideal situation, the number for this application would be 0.

4.2.3 Drift Estimation by Sensor Fusion Using EOT and SVM

Calibration drift estimation using EOT and SVM was evaluated using the eight scenarios (AMS4-AMS11) from the AMS dataset that were used previously for assessing AAKR and PCR algorithms. However, additional scenarios (AMS12-AMS16) were also evaluated to determine applicability to drift detection, when the measurement drift is due to pressure transmitter faults such as sensing line blockage.

The measurements from sensors Weed-d1q, Weed-d5s, Foxboro-d3i and Safir-d2r are fused using the EOT and SVM methods to estimate the output of sensor Rosemount-d4k, and the fusers are trained using measurements from Scenario 1. These sensors have different measurement units and ranges, as can be seen in Figure 13(a). Weed-d1q, Weed-d5s and Rosemount-d4k measure in units of inches of H_2O but have different ranges: namely, 0–850, 0–250, and 0–750, respectively. Foxboro-d3i measures in PSI units in a range of 0–50, and Safir-d2r measures in units of kPa in a range of 0–259. The measurements of Weed-d1q, Weed-d5s, and Rosemount-d4k overlap since they use the same units. Measurements of Safir-d2r are lower since units are different with lower range, and those of Foxboro-d3i are even lower due to different units and lower range. The measurements under Scenario 2 are shown in Figure 13(b), where the heater temperature is increased at the same times and by same amount as in Scenario 1. However, Rosemount-d4k is subjected to a calibration change that resulted in lowering its measurements compared to Weed-d1q, while other measurements remain close to their values in Scenario 1. The difference

between the measurements of Weed-d1q and Rosemount-d4k are used as a ground truth estimate of the drift in Rosemount-d4k measurements.



(a) training dataset: no drifts *(b) test dataset: drifted Rosemount-d4k*

Figure 11. Measurements for training dataset from Scenario 1 and a test dataset from Scenario 2.

Note that sensors have different units or ranges: Weed-d1q: 0–850 in. H_2O; Foxboro-d3i: 0–50 PSI; Rosemount-d4k: 0–750 in. H_2O; Safir-d2r: 0–259 kPa; and Weed-d5s: 0–250in. H_2O.

The performance of hetero-associative regressive information fusion method based on EOT and SVM regressions is assessed using RMSE of their sensor drifts estimates for Rosemount-d4k in Scenarios 1–20, which are shown in Figure *12*. Here, RMSE is expressed as a percent of the largest sensor reading of Rosemount-d4k around 200. For both, RMSE is under 2 percent in all scenarios, and EOT achieves lower RMSE overall. In the previous sections, AAKR and PCR methods are applied to Scenarios 2–9 (AMS4-11) that correspond to calibration errors. The EOT and SVM estimators are trained with measurements from Scenario 1, and RMSE for this case corresponds to the training error. Measurements from all other scenarios are used for testing the drift estimate for the Rosemount-d4k sensor. Scenarios 10, 14, and 20 do not involve introducing error into the Rosemount-d4k sensor, and yet their drift estimates have RMSEs comparable to other cases, albeit under 2 percent. These RMSE values are of the same order as variations in the measurements of the sensors used as input to EOT and SVM fusers.

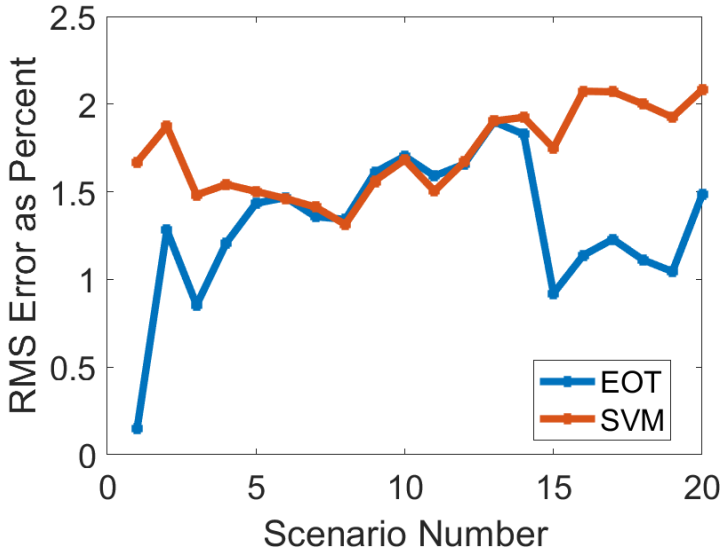
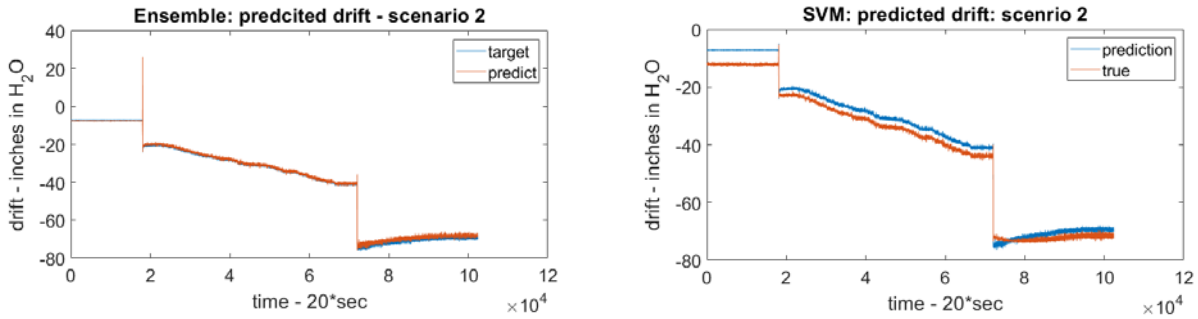


Figure 12. RMS error of sensor drift estimates of EOT and SVM methods.

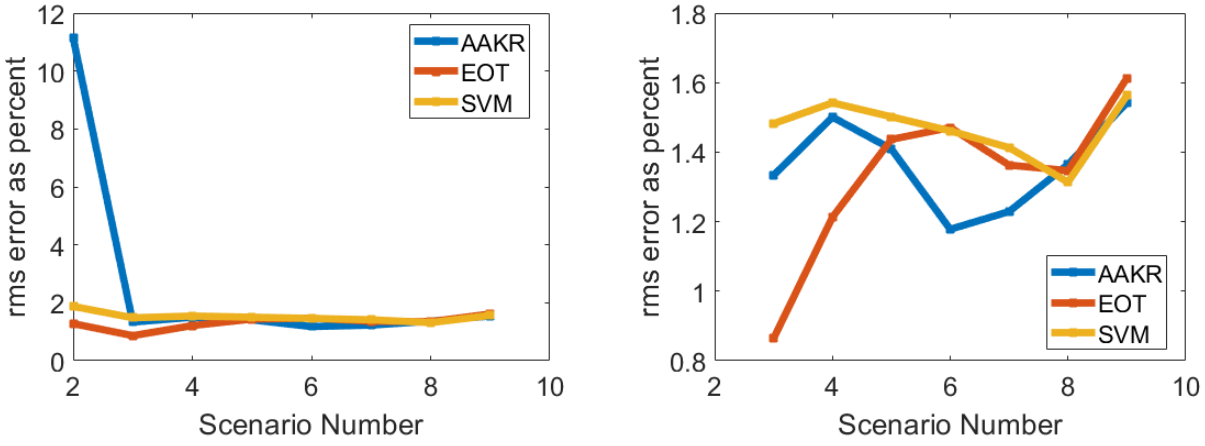
For illustrative purposes, the drift estimates for EOT and SVM methods are shown in Figure 15 for Scenario 2, in which the RMSE of the latter is higher. The drift estimate from SVM deviates from the true drift for most of measurements; in particular, it is less negative. The drift estimate for EOT is more accurate for almost all the measurements until the second temperature increase, after which it is less negative than the ground truth estimate.



(a) drift estimate of EOT method *(b) drift estimation of SVM method*

Figure 13. Sensor drift estimates of EOT and SVM methods for Rosemount-d4k differential pressure sensor.

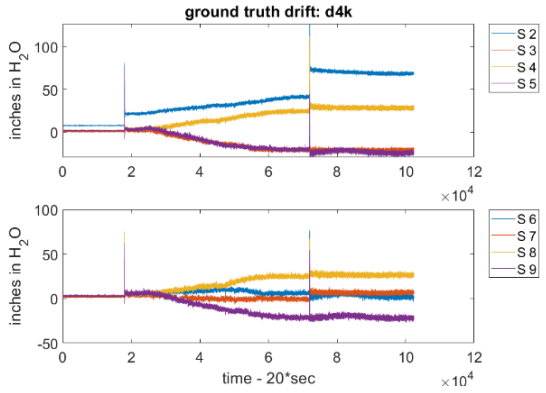
It is instructive to compare the drift estimates of these methods with those based on using the residuals of the AAKR method, which is applied to Scenarios 2–9 (AMS4-11) that correspond to calibration errors. A summary of RMSEs is shown in Figure 16. For Scenario 2, AAKR has a large error since the sensor drift is large, and drifted measurements are used in estimating the drift, as shown in Figure *14*(a). When Scenario 2 is excluded, RMSEs are comparable among the three, except for Scenario 3, and AAKR which has a lower RMSE than EOT in only 3 out of 9 scenarios.



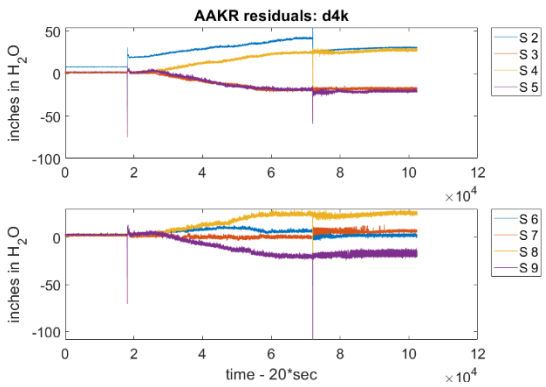
(a) Scenarios 2-9 *(b) Scenarios 3-9*

Figure 14. RMSE of AAKR, EOT and SVM drift estimates.

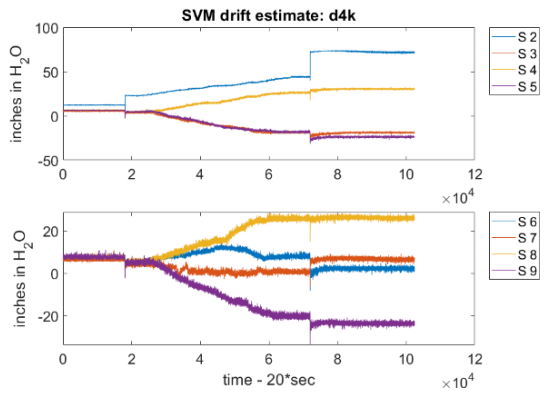
A qualitative characterization of the drift estimates can be gained by the collection of drift estimates from the three methods and the ground truth drift estimates shown in Figure *15*. For Scenario 2, the ground truth drift estimate exceeds 50 towards last third period, which is tracked by both EOT and SVM methods, whereas the AAKR residuals lead to much smaller deviations and hence higher RMSEs.



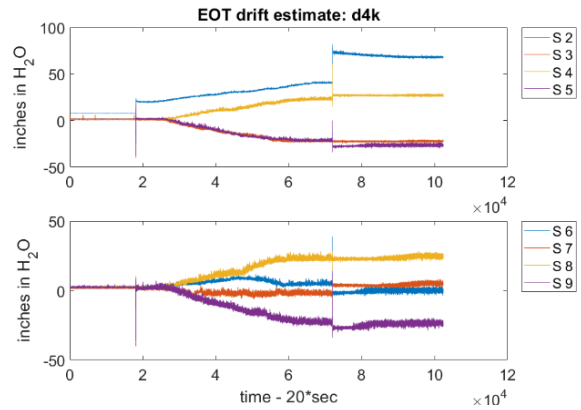
(a) Ground truth drift estimates



(b) AAKR drift estimates



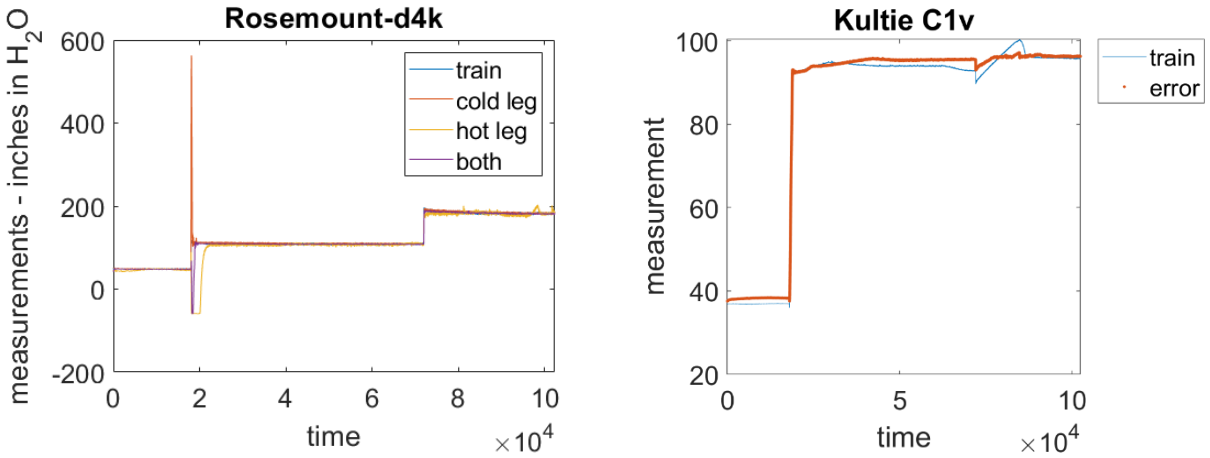
(c) SVM drift estimates



(d) EOT drift estimates

Figure 15. Drift estimates of three methods and ground truth estimates.

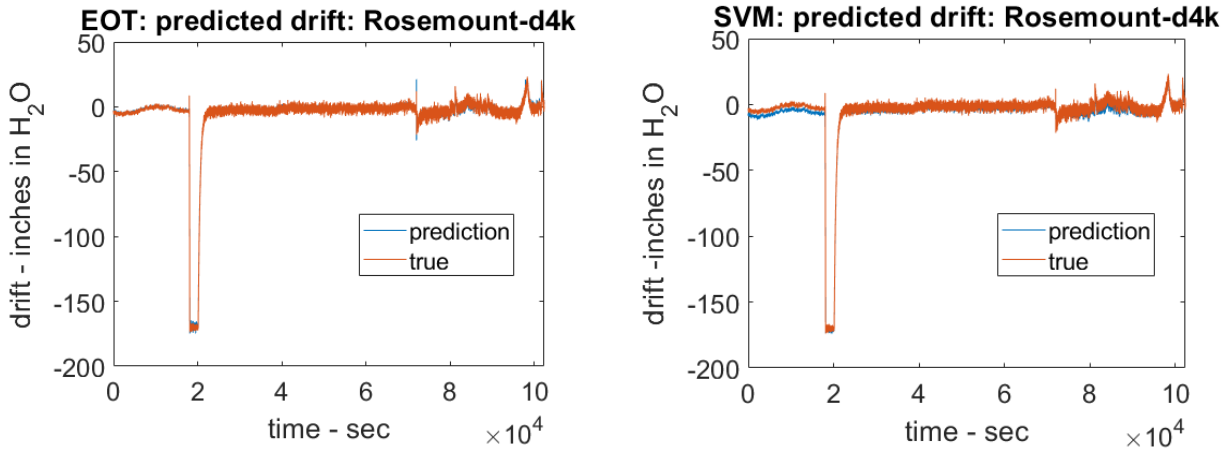
A more detailed look at Scenarios 10–14 (AMS12-16) is considered next. EOT and SVM fusers are trained using measurements from Scenario 10 shown in Figure 1(a). In Scenarios 11–14, various types of sensing line blockage were introduced into Rosemount-d4k differential pressure sensor and Kulite-c1v pressure sensor. In Scenarios 11–13, blocking errors were introduced in hot, cold, and both legs of the sensing lines for the DP sensor (Rosemount-d4k), and the measurements in 16 show both positive and negative measurement errors, as well as errors in transient and steady-state measurements. In particular, measurements from two other DP sensors (Weed-d1q and Weed-d5s) are close to each another, whereas measurements of Rosemount-d4k deviate from them in two different ways: specifically, significantly at the first temperature shift and somewhat slightly subsequently. In Scenario 5, sensing-line blockage was introduced in the cold leg sensing line for Kulite-c1v, and the measurements mostly result in steady-state errors, as shown in Figure 16(b).



(a) Rosemount-d4k traces: Scenarios 11-13 *(b) Kulite-c1v traces: Scenario 14*

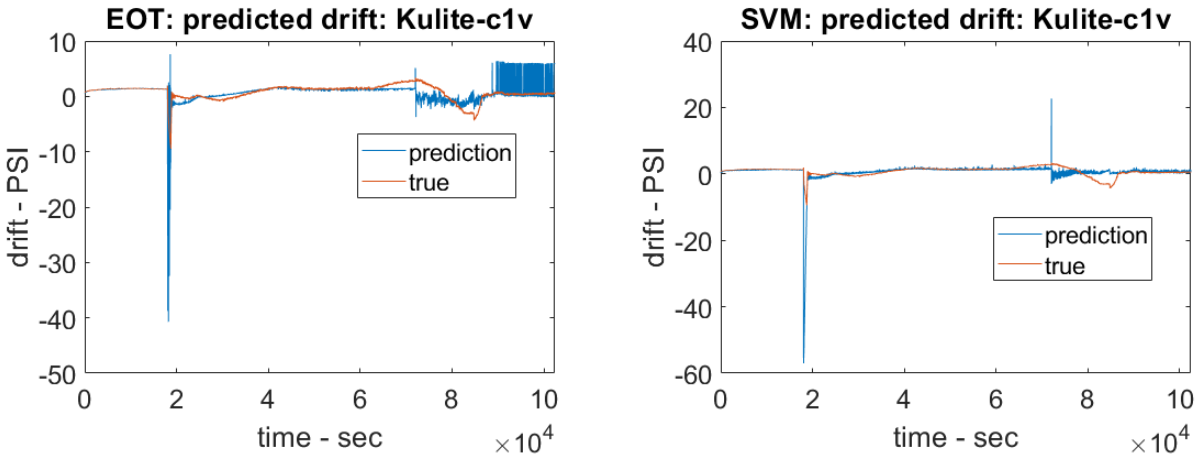
Figure 16. Measurement traces under blockage errors: scenarios 11–13 for Rosemount-d4k, and scenario 14 for Kulite-c1v.

For SVM and EOT regressions, measurements of Rosemount-d4k and Kulite-c1v constitute the dependent variables (in Scenarios 11–13 and 14, respectively) and the independent variable is a 4–dimensional vector corresponding to measurements of four other DP sensors. Overall, results indicate that positive and negative sensor errors are captured by both methods in scenarios ranging from slow and small to rapid and large faults. For illustration, drift estimates based on their outputs are shown in Figure *17* for Scenario 11 for Rosemount-d4k and in Figure *18* for Scenario 14 for Kulite-c1v. The drift in Scenario 11 in Rosemount-d4k is identified by both, and EOT's estimate is closer during the initial part and toward the end of the time period. The drifts in Scenario 14 are subtler and were tracked well by both methods in the middle of the experiment, but they were overestimated during the first temperature change. The overall trends are better tracked by EOT than by SVM, and both exhibited significant variations toward the end as a result of similar variations in the measurements of other sensors used as input to regression estimation.



(a) EOT drift estimate for Rosemount-d4k *(b) SVM drift estimate for Rosemount-d4k*

Figure 17. Drift estimates of EOT and SVM for Rosemount-d4k for Scenario 11.



(a) EOT drift estimate for Kulite-c1v *(b) SVM drift estimate for Kulite-c1v*

Figure 18. Drift estimates of EOT and SVM for Kulite-c1v for Scenario 14.

The RMSEs of estimated sensor faults for five scenarios are shown in Figure 19 expressed as a percent of the maximum sensor reading, which shows that estimation error of the EOT method is lower overall. The estimation error is within 1.44% and 1.63% for EOT and SVM methods, respectively.

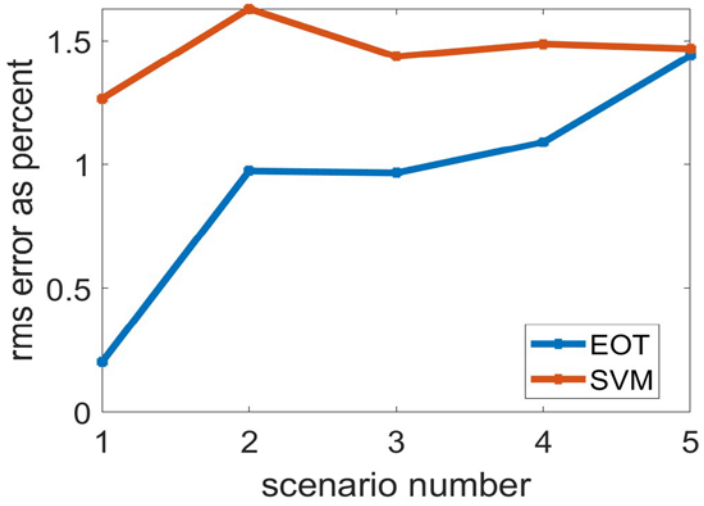


Figure 19. Root mean square error of EOT and SVM fusers.

Overall, the dataset featured 20 scenarios. While the method for introducing drifts via calibration changes, blockages, and others into the loop may differ from scenario to scenario, the basic method can be applied to all scenarios. This can test the robustness of the method, as the different methods should affect the measurements in different ways.

It is worth noting that the analyses discussed above demonstrate the effectiveness of the technique for drift estimation. Approaches for detecting the onset of drift from these estimates are being developed and will be discussed in future reports.

4.2.4 Generalization Error Equations for SVM, EOT and AAKR

Generalization can be considered the most important attribute of evaluating a learned model. In statistical learning theory, generalization performance of a learning method relates to its prediction capability on a set of unseen samples drawn from the distribution same as that of the training set [41]. Assessment of this performance is extremely important in practice, since it guides the choice of learning method or model, and gives a measure of the quality of the ultimately chosen model. From a mathematical point of view, the process of learning is to find a function $f(\boldsymbol{x})$ through a training set $\mathbb{X} = \{(\boldsymbol{x_i}, t_i)\}_{i=1}^{N} \subset \mathcal{R}^n \times \{0,1\}^c$ such that $f(\boldsymbol{x})$ can approximate the objective function $F(\boldsymbol{x})$ both at training cases and unseen cases. The difference between $F(\boldsymbol{x})$ and $f(\boldsymbol{x})$ is called *generalization error*.

This section presents the generalization error equations that characterize the performance of sensor drift estimates on future measurements using the distribution-free machine learning formulation. By using smoothness (bounded Lipschitz constant) and non-smoothness (fixed number of finite jumps) properties, they guarantee that the generalization error is bounded by a specified parameter—called the *precision*, with probability—called the *confidence*. The generalization error equations establish that, for these methods, the MSE of the drift estimate is bounded by the precision parameter with the probability given by the confidence parameter that improves with the size of the training data.

More generally, these generalization error equations provide critical performance insights into these methods:

1. The existence of these equations in this form shows that the underlying estimation problem is solvable in principle by machine learning methods. These equations may not exist for certain complex problems, such as estimating the state transition map from chaotic iterates.
2. The form of a generalization equation provides important qualitative information, which in this case shows that increasing sample size and decreasing dimensionality each independently leads to

improved confidence. This clear separation is a consequence of the smoothness of underlying primary coolant system dynamics.

3. By using the detailed parameter values, practical confidence levels can be obtained.

The problem of drift estimation can be cast as a regression estimation problem by identifying a drift estimation method by its regression estimate $\hat{f}$. By mapping the sensor measurements to the independent variable X and the drift to the dependent variable Y, we obtain $\hat{f}(X)$ as an estimate of drift. Here, X and Y are random variables distributes according to $P_{X,Y}$, which is typically complex and unknown since it encompasses various sensing and process errors. For a regression estimate f, the expected error is defined as $I(f) = \int (f(X) - Y)^2 d P_{X,Y}$. Let f^* denote the optimal regression that minimizes the expected error. The regression estimate $\hat{f}$ is computed using a training sample $(X_1, Y_1), (X_2, Y_2), \ldots, (X_l, Y_l)$, and its generalization error equation is expressed as

$$P_{Y,X}^{l}\left[I(\hat{f}) - I(f^*) > \epsilon\right] < \delta, \tag{3}$$

which implies that the error of estimated regression is within precision $\boldsymbol{\varepsilon}$ of the optimal regression function with confidence probability of $\mathbf{1} - \boldsymbol{\delta}$, irrespective of the underlying joint probability distribution $\boldsymbol{P_{X,Y}}$, where $\boldsymbol{\varepsilon > 0}$ and $\mathbf{0 < \delta < 1}$. A further discussion of the derivation and meaning of generalization error equations can be found in Appendix A. The results for the three methods are presented and some practical implications of the generalization error equations are briefly discussed here.

SVM: smoothness property. Smooth regression estimates are generated by several machine learning methods, including sigmoidal neural networks, radial basis functions, potential functions, and the SVM method with Gaussian kernels. An important subclass of smooth functions are Lipschitz continuous functions for which $|f(x) - f(x + \partial x)| \leq L\|\partial x\|$, where L is a Lipschitz constant. Differentiable functions satisfy this condition, wherein the maximum derivative can be used as L. This property, combined with the boundedness of domain variables such as pressure, temperature, and the flow rate of the coolant system, provide generalization bounds that explicitly show their dependence. We consider a generic form of SVM regression given by $f_{SVM} = \sum_{i=1}^{g} (\alpha_i - \alpha_i^*) exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right) + b$ with g Gaussian kernels[3] where the parameters are bounded such that $\alpha_i, \alpha_i^* \in [0, C]$. The magnitude of the derivative of kernel $e^{-\sum_{j=1}^{d}(x_i - x_j)^2}$ is upper bounded by 1. An estimate for Lipschitz constant $L = 1$ is obtained for f_{SVM}. Then the following expression for δ of f_{SVM} is obtained:

$$\delta_{SVM} = 8\left(\frac{32C}{\varepsilon}\right)^{2d} e^{-\varepsilon^2 l/512}, \tag{4}$$

As derived in Appendix A following the derivation in [26] in which d is the dimension of the hyperplane of the SVM regression estimate. In general, increasing d has the effect of reducing the optimal error $I(f^*)$, but it has a negative effect on generalization property as indicated by above expression. The first part expression for δ depends on the SVM parameters d and C, and the second part depends on the sample size l with opposite effects: increasing the sample size improves the confidence via the exponential function, whereas increasing the dimensionality d decreases the confidence again via an exponential function. Thus, ensuring high confidence guards against overfitting the training data by increasing the dimensionality of SVM regression.

4.2.4.1 EOT: non-smooth property. Machine learning methods also employ non-smooth regression functions such as ensemble tree methods [29] and regression trees [30]. In practice, the variables are bounded, and the learned functions have a finite (often small) number of jumps, which leads to their bounded finite total variation $V < \infty$.[30] In this case, the confidence function is given by $\delta_{EOT} = 16\left(\frac{4l}{\varepsilon^2}\right)^{d \log_2(2el/(d\varepsilon))} e^{-\epsilon^2 l/2048}$, with the shattering index $d = \left(1 + \frac{128V}{\varepsilon}\right)$ derived in Appendix A. Let

f_{EOT} be the regression tree estimate consisting of N_L leaves and let B be the upper bound for dependent variable Y, for example, the highest differential pressure. When viewed in terms of function of X, each leaf corresponds to a region with a variation, at most $2B$. By accounting for all leaves, the total variation of f_{EOT} is upper bounded by $2BN_L$. Then we obtain the following confidence function for f_{EOT}:

$$\delta_{EOT} = 16\left(\frac{4l}{\varepsilon^2}\right)^{\left(1+\frac{256BN_L}{\varepsilon}\right)\log_2(2el/(\varepsilon+256BN_L))} e^{-\varepsilon^2 l/2048} \quad (5)$$

by using $V = 2BN_L$ in the above formula. As in the case of SVM, the effects of properties of EOT regression and the sample size can be seen as independent.

4.2.4.2 AAKR: auto-associative property

AAKR is a nonparametric model that makes prediction by comparing a query to past input examples. Past input examples form the memory vector, and they are the parameters of the prediction step together with the bandwidth of the Gaussian kernel. To maintain the AAKR as a Lipschitz continuous function, the kernel bandwidth needs to be *a priori* fixed. With this form, memory-vector parameters can be used to find the Lipschitz constant. Memory-vector parameters are bounded by $X \in [0,1]$. The distance matrix also can be bound with the memory vector, which helps to determine the Lipschitz constant. The Lipschitz constant for AAKR is obtained as follows:

$$\mathrm{L} = \frac{4 + \mathrm{h}^2}{\mathrm{h}^2 \mathrm{n_m} \mathrm{e}^{-\frac{2\mathrm{p}}{\mathrm{h}^2}}} \quad (6)$$

Then the following confidence function for f_{AAKR} is obtained as derived in Appendix A:

$$\delta_{\mathrm{AAKR}} = 8\left(\frac{128 + 32\mathrm{h}^2}{\varepsilon \mathrm{h}^2 \mathrm{n_m} \mathrm{e}^{-\frac{2\mathrm{p}}{\mathrm{h}^2}}}\right)^{\mathrm{n_m p}} \mathrm{e}^{-\varepsilon^2 \mathrm{l}/512} \quad (7)$$

Again, as in the case of SVM, the effects of properties of AAKR solution and the sample size can be seen as independent.

4.3 Discussion

Table 2 summarizes the advantages and disadvantages of investigated algorithms. If the algorithm can be used as an auto-associative model, a group of features is selected as both inputs and outputs. Therefore, within the group, no subselection is needed. This provides an advantage for scalability. If the method is robust when a drifting or faulty sensor is included in the inputs, then the accuracy of the prediction is high. If the method has low computing cost, it benefits for scalability and near real-time detection.

Table 2. Summary of the advantages and disadvantages of the algorithms investigated.

Method	Auto-associative?	Robustness	Training cost	Inference cost
AAKR	Yes	Medium	Medium	Medium
PCR	Yes	High	Low	Medium
EOT	No	High	Medium	Low
SVM	No	High	Medium to High	Low

5. CONCLUSIONS

Online monitoring (OLM) has the potential for assisting in calibration-interval extension in commercial NPPs and directly helping reduce the overall operations and maintenance costs. The history of OLM in NPP applications has pointed to the need for addressing key questions related to accuracy and setpoint uncertainty before the technology can be deployed in TS applications. Alternative approaches to deploying OLM for performance monitoring may be possible; here again the need is to address accuracy of sensor drift detection due to aging or sensor faults.

Advances in data analytics provide a potential pathway towards addressing these concerns. Several algorithms are available that may be adapted for this purpose, including PCR, SVM, and EOT. In most scenarios, these algorithms may be used to fuse information from multiple sensors to estimate (model prediction) the output of one or more sensors. While such a data-driven approach has limitations, especially in generalization of the model outputs to test data, it also has the advantage of speed and can embed unique operational characteristics and relationships between measurements at different locations of the specific plant. By contrast, OLM methods that use physics-based models require model tuning to capture such individualized behavior, which may be a challenge. For the data-driven methods discussed in this report, generalization error bounds may also be calculated theoretically and provide an upper bound on the uncertainty.

The algorithms were evaluated using existing datasets from a laboratory-scale flow loop with simulated sensor faults and sensor drift. The algorithms were compared against a standard OLM algorithm—AAKR—on several counts: accuracy, sensitivity, and detection performance. The algorithms tested in this study appeared to compare favorably against AAKR on this dataset, though additional evaluations against plant operational data are needed to fully characterize the performance of these algorithms. Ongoing research includes evaluating additional datasets, using data from test reactors and plant operational data.

Approaches for using the algorithm outputs for recalibrating the sensor online will be examined next. Methods for accelerating the deployment of OLM in the fleet are being evaluated under a different research program by other organizations; this project will engage with these organizations to better understand barriers to deployment and update the research plan to address these barriers in the next phase of research.

6. REFERENCES

[1] NRC, *Safety Evaluation of EPRI Topical Report TR104965, Online Monitoring of Instrument Channel Performance,* ML3734509, 2000.

[2] P. Goffin, B. Shumaker, A. Hashemian and G. Morton, "Online Monitoring for static and dynamic performance verification of I&C systems at Sizewell B nuclear power station," *Proc. NPIC&HMIT 2017,* San Francisco, CA, June 2017, pp. 1698–1705.

[3] Bernhard Schölkopf, Christopher JC Burges, and Alexander J. Smola, eds. *Advances in kernel methods: support vector learning.* MIT press, 1999.

[4] J. B. Coble, R. M. Meyer, P. Ramuhalli, L. J. Bond, H. Hashemian, B. Shumaker and D. Cummins (2012). "A Review of Sensor Calibration Monitoring for Calibration Interval Extension in Nuclear Power Plants." PNNL-21687.

[5] T. Quinn, J. Mauck, et al, *Digital Sensor Technology*, INL/EXT-13-29750, July 2013.

[6] U.S. NRC, *Standard Review Plan for the Review of Safety Analysis Reports for Nuclear Power Plants: LWR Edition – Instrumentation and Controls,* NUREG-0800, Chapter 7, Rev 6, August 2016.

[7] E. Davis, and B. Rasmussen. *Online Monitoring of Instrument Channel Performance: Volume 3: Applications to Nuclear Power Plant Technical Specification Instrumentation.* No. 1007930. EPRI (US), 2004.

[8] *Investigation of Response Time Testing Requirements,* EPRI, Palo Alto, CA: 1991. NP-7243.

[9] U.S. NRC, *Standard Review Plan for the Review of Safety Analysis Reports for Nuclear Power Plants: LWR Edition – Instrumentation and Controls,* NUREG-0800, Chapter 7, BTP 7-13 Rev 6, *Guidance on Cross-Calibration of Protection System Resistance Temperature Detectors*, August 2016.

[10] *On-Line Monitoring of Instrument Channel Performance,* EPRI, Palo Alto, CA: 1998. TR-104965.

[11] J. W. Hines, *On-Line Monitoring for Calibration Extension: An Overview and Introduction*: An overview of NUREG/CR-6895, ML091400211.

[12] EPRI, "On-Line Monitoring Cost Benefit Guide," Final Report 1006777, 2003.

[13] J. Hines and R. Seibert, *Technical Review of On-Line Monitoring Techniques for Performance Assessment: Volume 1: State-of-the-Art,* NUREG/CR-6895, 2006.

[14] EPRI (2006), TR-1013486 *Plant Application of On-Line Monitoring for Calibration Interval Extension of Safety-Related Instruments: Volume 1*, EPRI, Palo Alto, CA, and British Energy Group PLC, Suffolk, UK.

[15] EPRI (2006), TR-1013486 *Plant Application of On-Line Monitoring for Calibration Interval Extension of Safety-Related Instruments: Volume 2*, EPRI, Palo Alto, CA, and British Energy Group PLC, Suffolk, UK.

[16] EPRI (2007), TR-1015173 *Plant Application of On-Line Monitoring for Calibration Interval Extension of Safety-Related Instruments: Volume 3*, EPRI, Palo Alto, CA, and British Energy Group PLC, Suffolk, UK.

[17] U.S. NRC Draft Safety Evaluation of TSTF-569 Revision 1, *Revise Response Time Testing Definition*, May 2019, ML19017A220.

[18] P. Ramuhalli, R. Tipireddy, J. Coble, B. Shumaker, M. Lerchen, A. Nair, and H. Hashemian (2016). *Development of Fault Detection and Virtual Sensing Methodologies for Robust Online Monitoring* (PNNL-25382).

[19] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part 1*. George Mason University, 2001.

[20] Steven M. Kay, *Fundamentals of statistical signal processing.* Prentice Hall PTR, 1993.

[21] V. N. Vapnik, *Statistical Learning Theory*, John-Wiley and Sons, New York, 1998.

[22] J. W. Hines, et al., *Technical Review of On-Line Monitoring Techniques for Performance Assessment: Volume 1: State-of-the-Art, Volume 2: Theoretical Issues, Volume 3: Limiting Case Studies*, NUREG/CR-6895, 2006.

[23] Jianping Ma. "Methods and Systems for Fault Diagnosis in Nuclear Power Plants," University of Western Ontario (2015).

[24] Daniel S. Rizzuto and Michael J. Kahana. "An autoassociative neural network model of paired-associate learning." *Neural Computation* 13, no. 9 (2001): 2075–2092.

[25] J. Wesley Hines. "Robust distance measures for on-line monitoring." US Patent No. 8,311,774. 13, Nov. 2012.

[26] N. S. V. Rao, "Simple sample bound for feedforward sig-moid networks with bounded weights," *Neurocomputing,* vol. 29, pp. 115-122, 1999.

[27] N.S. V. Rao, P. Ramuhalli, C. Greulich, S. M. Cetiner, and P. Devineni, "Sensor Error Estimation for Reactor Coolant System with Generalization Bounds," in *ANS Winter Conference*, 2019.

[28] Nageswara SV Rao, David B. Reister, and Jacob Barhen. "Information fusion methods based on physical laws." IEEE transactions on pattern analysis and machine intelligence 27, no. 1 (2005): 66–77.

[29] Kagan Tumer and Joydeep Ghosh. "Error correlation and error reduction in ensemble classifiers." *Connection Science* 8, No. 3-4 (1996): 385–404.

[30] Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 2009.

[31] L. Breiman J. H. Friedman, R. A. Ohlsen, and C. J. Stone. "Classification and Regression Trees. Wadsworth & Brooks/Cole." *Advanced Books & Software* (1984).

[32] N. S. V. Rao, P. Ramuhalli, C. Greulich, S. M. Cetiner and P. Devineni, Sensor Drift Estimation for Reactor Systems by Fusing Multiple Sensor Measurements, 2019 IEEE Nuclear Science Symposium, October 26–November 2, 2019, Manchester, UK. Pub ID 127360.

[33] Mohamed-Faouzi Harkat, Salah Djelel, Noureddine Doghmane, and Mohamed Benouaret. "Sensor Fault Detection, Isolation and Reconstruction Using Nonlinear Principal Component Analysis." *International Journal of Automation and Computing* 4, No. 2 (2007): 149–155.

[34] Ian T. Jolliffe, "A note on the Use of Principal Components in Regression." *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 31.3 (1982): 300–303.

[35] R. Dennis Cook. “Detection of Influential Observation in Linear Regression.” *Technometrics* 19.1 (1977): 15–18.

[36] Arthur E. Hoerl and Robert W. Kennard. “Ridge Regression: Applications to Nonorthogonal Problems.” *Technometrics* 12.1 (1970): 69–82.

[37] Jerome H. Friedman, "Stochastic Gradient Boosting." *Computational Statistics and Data Analysis* 38.4 (2002): 367–378.

[38] Andy Liaw and Matthew Wiener. “Classification and Regression by Random Forest.” *R News* 2.3 (2002): 18–22.

[39] J. W. Hines et al. *Technical Review of On-Line Monitoring Techniques for Performance Assessment* (NUREG/CR-6895) Vol. 2, “Theoretical Issues.” (2008).

[40] P. Ramuhalli, R. Tipireddy, J.B. Coble, A.M. Nair, S. Boring, B. Shumaker, and M.E. Lerchen. 2017. *Robust Online Monitoring Technology for Calibration Assessment of Transmitters and Instrumentation;* PNNL-26919, Pacific Northwest National Laboratory, Richland, WA.

[41] T. Hastie, R. Tibshirani, J. Friedman, “The Elements of Statistical Learning: Data Mining, Inference, and Prediction,” Springer Series in Statistics, Second Edition, New York (2001).

Appendix A

Generalization Theory and Application to Data Analytics Methods

The generic problem of learning is finding a relationship between two variables that are related with a joint probability distribution $P_{x,y}$. These variables can be vectors with dimension d in real vector space and can be expressed as $X \in \mathbb{R}^d$ and $Y \in \mathbb{R}^d$. The joint probability distribution is not known for most of the cases; therefore, X is approximated to Y with an estimate function f. In regression problems, f is a regression function such that $f(X)$ is an estimate of observations Y. The f is a subset of hypothesis space, $\mathcal{F}$ and hypothesis space is the set of all possible functions that can return $X \rightarrow Y$.

A good estimate function should have low error, and it can be quantified by the expected error. The expected error measures the loss averaged over the unknown distribution and can be calculated with Equation (8)(9). A good estimate function should have small expected cost

$$I(f) = \int C(f(X), Y)\, dP_{X,Y}, \tag{8}$$

where C is the cost function, the measure of how good the approximation is and what is the cost of approximating X to Y. For regression problems, two most used cost functions are square loss, $(f(X) - Y)^2$ and absolute value loss, $|f(X) - y|$. In hypothesis space, there is an expected best function f^* that minimizes the expected error. However, since $P_{X,Y}$ is not known for most of the cases, f^* cannot be computed precisely.

Regression algorithms are applicable to samples with finite size. For an l-sample dataset or a training set, $S = \{(x_1, y_1), \ldots, (x_l, y_l)\}$ consists of n samples drawn independent and identically from the $P_{x,y}$. A learning algorithm is an algorithm that takes the dataset and selects an estimate function from the hypothesis space. The error of estimate function on dataset is quantified by the empirical error and can be calculated with Equation (9). In hypothesis space, there is a suitable function $\hat{f}$ that minimizes the empirical error and $\hat{f}$ can be calculated for a dataset.

$$I_{emp}(f) = \frac{1}{l} \sum_{i=1}^{l} C(f(X_i), Y_i). \tag{9}$$

A natural requirement for f is distribution independent generalization which implies that as the amount of training data increases, the training error for the solution must converge to the expected error. A solution is predictive if it satisfies the condition of Equation (10), and this means that when the dataset is large, the empirical error is a good proxy for the expected error:

$$\forall P \lim_{n \to \infty} \left| I(f) - I_{emp}(f) \right| = 0. \tag{10}$$

It is practically impossible to draw infinite samples; therefore, the empirical error will be different from the expected error. The empirical error is minimized by $\hat{f}$, and it is important to assess how $\hat{f}$ is close to f^* for a finite dataset. Vapnik's generalization theory ensures that the expected error of $\hat{f}$ is within ε of the expected error of f^* with probability of $1 - \delta$, irrespective of the $P_{X,Y}$, where δ is the confidence parameter. This condition can be expressed by Equation (11):

$$P_{Y,X}^{l} \left[I(\hat{f}) - I(f^*) > \epsilon \right] < \delta_M(\boldsymbol{F}_M, \epsilon, l). \tag{11}$$

The error in the square bracket, $I(\hat{f}) - I(f^*)$, is called the generalization gap. In these scenarios, minimization of empirical error is not assured, and hence following version of Equation (11) is used, as given by

$$\mathrm{P}_{\mathrm{Y,X}}^{\mathrm{l}}\left[\mathrm{I}(\hat{\mathrm{f}}) - \mathrm{I}(\mathrm{f}^*) > \epsilon + \hat{\epsilon}\right] < \widehat{\delta_{\mathrm{M}}}(\mathbf{F}_{\mathrm{M}}, \epsilon, \hat{\epsilon}, \mathrm{l}), \tag{12}$$

where $\hat{\epsilon}$ is the training error associated with computing $\hat{f}$. In this case, for Rosemount-d4k using EOT, we have $\hat{f} = f_{EOT}$, $F_M = F_{EOT}$ and $\hat{\epsilon} = 0.3947$, and using SVM, we have $\hat{f} = f_{SVM}$, $F_M = F_{SVM}$, and $\hat{\epsilon} = 2.48$. To simplify presentation, we estimate $\delta_M(.)$ as a function of ϵ, l and properties of F_M based on smoothness and non-smoothness properties.

Deriving a generalization bound for Equation (12) and relating error parameter to confidence parameter is an important step in assessment of machine-learning algorithms. If an algorithm has a theoretical generalization bound, it can be regarded as a proof that the algorithm is predictive, and the generalization error can be related to the sample size of the dataset. The generalization equation in Equation (14) is expressed in the following generic form for machine-learning method M,

$$\mathrm{P}_{\mathrm{Y,X}}^{\mathrm{l}}\left[I(\hat{f}) - I(f^*) > \epsilon\right] < \mathrm{G}_{\mathrm{M}}(\boldsymbol{F}_M, \epsilon, \mathrm{l})\mathrm{e}^{-\mathrm{g}_{\mathrm{M}}(\epsilon, \mathrm{l})}, \tag{13}$$

where the functions $G_M(F, \epsilon, l)$, and $g_M(\epsilon, l)$ are obtained based on the specific properties of the method. Note that the method-specific function can and does change with different implementations of the same method. In the derivation of the bounds, it is important to understand the specific implementation of the method employed in your analysis. Typically, $g(\epsilon, l)$ increases in l, often linearly, and $G_M(F, \epsilon, l)$ is either fixed or increases more slowly than exponentially; consequently, the right-hand term decreases in l and can be made to match a specified value for δ_M for a large enough sample size. Under this condition, the generalization error is bounded as $I(\hat{f}) < I(f^*) + \epsilon$ with probability $1 - \delta_M$.

In order to find the confidence parameter in generalization equation, we adopted Rao's method[26]. Equation (14)(15) has the following form:

$$P_{Y,X}^{l}\left[I(\hat{f}) - I(f^*) > \epsilon\right] < 8N_{\infty}(\varepsilon/32, \boldsymbol{F}_M)e^{-\varepsilon^2 l/512}. \tag{14}$$

N_∞ is called the *covering number*, which represents the minimum number of spherical balls of radius ε needed to cover the set $\boldsymbol{F}$. In order to calculate the covering number, the Lipschitz property can be used. Lipschitz continuous functions are functions that have a limited growth rate. If a function satisfies the Lipschitz property, then there exists a constant such that every pair of points of this function is connected with a line that has a smaller slope than this constant. This constant is called the *Lipschitz constant* and is denoted by L. Considering that f can get values within the bounds of weights, using this with the Lipschitz constant provides a cover size estimate and can be used as the covering number. Covering numbers for SVM and AAKR are calculated using this property. For an estimate function $f_w \in \boldsymbol{F}_W$ where $w \in [-W, W]^d$, the Lipschitz property can be written as follows,

$$|f_u(x) - f_v(x)| \leq L\|u - v\|_{\infty}, \tag{15}$$

where u and v are two different weights with $u, v \in [-W, W]^d$ and $\|u - v\|_\infty = \max_{i=1,\dots,d} |u_i - v_i|$. For a uniform grid size of $(2W/\varepsilon)^d$, there exists w_1 in the cover that satisfies the condition $\|w - w_1\|_\infty \leq \varepsilon$. Using the Lipschitz property, we have $|f_w(x) - f_{w_1}(x)| \leq L\varepsilon$ for all x. Therefore, the covering number for this case is given by

$$N_{\infty}(\varepsilon, \boldsymbol{F}_W) \leq (2LW/\varepsilon)^d. \tag{16}$$

Generalization of SVM

For the dataset $\{(x_1, y_1), \dots, (x_l, y_l)\}$, the goal of SVM regression is finding the function $f(x)$ that for each input data x_i, has at most ε deviation from the output data y_i. The input space with dimension d is denoted by $X \in \mathbb{R}^d$. The estimate function is given in Equation 18, where $w \in \mathbb{R}^d$ is the weight with dimension d, and $b \in \mathbb{R}$ is the bias that is a scalar number. The SVM algorithm aims to minimize the norm $\|w\|^2$ subject to $|y_i - \langle w, x_i \rangle - b| \leq \varepsilon$:

$$f(x) = \langle w, x \rangle + b. \tag{17}$$

However, within the distribution of data, there will be pairs (x_i, y_i) which reside beyond the ε band. These points are called *slack points* and are represented with ξ. Slack points are added to the minimization process with a parameter C which determines the tradeoff between flatness of f and toleration of points beyond ε. The new minimization problem now considers the slack points and represented with Equation (18):

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*). \tag{18}$$

It has been shown that such a minimization problem can be solved with dual formulation. The constructed Lagrange function has saddle points with respect to dual variables in the solution. The Lagrange function is given by,

$$\begin{aligned} L := \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) - \sum_{i=1}^{l}(\eta_i\xi_i + \eta_i^*\xi_i^*) \\ - \sum_{i=1}^{l}\alpha_i(\varepsilon + \xi_i + \langle w, x_i \rangle + b - y_i) \\ - \sum_{i=1}^{l}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i \rangle - b) \end{aligned} \tag{19}$$

where $\alpha_i, \alpha_i^*, \eta_i, and\ \eta_i^*$ are Lagrange multipliers. If we apply saddle-point condition, the partial derivatives of L with respect to primal variables are set to zero. The form of the dual-optimization problem is now given by Equation (20). It must be noted here that the summation of $(\alpha_i - \alpha_i^*)$ over l is subject to 0, and the Lagrange multipliers can take values between zero and C, $\alpha_i, \alpha_i^* \in [0, C]$.

$$-\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)\left(a_j - a_j^*\right)\langle x_i, x_j \rangle - \varepsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*). \tag{20}$$

After implicit nonlinear mapping with kernels, the support vector expansion is given in Equation (21). A common kernel is Gaussian radial basis function kernel which is given in Equation (22).

$$w = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\Phi(x_i),\ thus\ f(x) = \sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)k(x_i, x_j) + b, \tag{21}$$

$$k(x_i, x) = exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right). \tag{22}$$

All the steps of SVM regression can be written in a single function given by Equation (23). The kernel bandwidth, σ, is *a priori* fixed, and the learning problem deals with computing a suitable w, which is finding the α_i and α_i^*; therefore, the weight vector has a dimension of $2d$. The hypothesis space then can be written as $\mathcal{F}_{W,\sigma} = \left\{f_{w,\sigma} : w \in [0, C]^{2d}\right\}$, and an estimate function f is chosen from the hypothesis space.

$$f_{w,\sigma}(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i{}^*)exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right) + b. \tag{23}$$

In order to find the Lipschitz constant, we need to find bounds of the partial derivatives of $f_{w,\sigma}$ with respect to w. The partial derivatives and their bounds are given by

$$\begin{aligned} \frac{\partial f_{w,\sigma}}{\partial \alpha_i} &= exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right) \leq 1 \\ \frac{\partial f_{w,\sigma}}{\partial \alpha_i{}^*} &= \left|exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right)\right| \leq 1. \end{aligned} \tag{24}$$

The bounds on partial derivatives show that the Lipschitz constant equals 1 since the exponential term is always less than 1, independent of the input value. This implies that a uniform grid of size $(C/\varepsilon)^{2d}$ satisfies the Lipschitz property. Therefore, the covering number for SVM is given by

$$N_\infty\left(\varepsilon, \mathcal{F}_{W,\sigma}\right) \leq (C/\varepsilon)^{2d}. \tag{25}$$

Implementing the covering number for SVM into Equation (12) gives us the generalization equation for SVM,

$$P_X[I(f_{\hat{w}}) - I(f_{w^*}) > \varepsilon] \leq 8\left(\frac{32C}{\varepsilon}\right)^{2d} e^{-\varepsilon^2 l/512} \tag{26}$$

Generalization of EOT

EOT is a predictive model which uses combination of multiple regression trees with associated weights. A regression tree is a tree-shaped structure to represent a recursive partition. The end nodes of the trees are called leaves, and they represent the partition of the model which applies in that node only. The tree starts with the root node, and branching is done by asking a sequence of questions about the features. The branches are answers to the previous questions, and the next question to ask depends on the answers on that branch.

Since the EOT is a non-smooth method, the Lipschitz property cannot be used to find covering numbers. Instead, a covering-number estimate based on total variation[37] is used. For a function set of $\boldsymbol{F}$ with having total variation at most V, the *fat* shattering number is given by

$$fat_{\boldsymbol{F}}(\gamma) = 1 + \left[\frac{V}{2\gamma}\right]. \tag{27}$$

The shattering means that a function f can shatter a set of points if, for every possible training set, there exists a value of f that gets no training error. For a function set of $\boldsymbol{F}$ and sample set $S = \{x_1, x_2, \dots, x_l\}$, S is $\gamma - shattered$ by $\boldsymbol{F}$ if there exits real numbers $r_1, r_2, \dots, r_l$ and a function f_b in $\boldsymbol{F}$ that satisfies $f_b(x_i) \geq r_i + \gamma$ if $b_i = 1$ and $f_b(x_i) \leq r_i - \gamma$ if $b_i = 0$ for all samples. Then $fat_{\boldsymbol{F}}(\gamma)$, the γ-fat-shattering dimension of $\boldsymbol{F}$, is the largest set γ-shattered by $\boldsymbol{F}$, where γ is the width of shattering.

A hypothesis class $\boldsymbol{F}$ is a set of real functions from a domain X to the bounded interval of $[0, W]$. The complexity of $\boldsymbol{F}$ is given by the growth function, and growth function with its bounds are given by Equation (30). The first inequality follows from the Binomial Theorem, and the second inequality follows from Euler's inequality. d is the fat-shattering number of $\boldsymbol{F}$ with a width of $\varepsilon/4$, $d = fat_{\boldsymbol{F}}(\varepsilon/4)$

$$y = \sum_{i=1}^{d} \binom{l}{i} \left(\frac{2B}{\varepsilon}\right)^{i} < \left(\frac{2B}{\varepsilon}\right)^{d} \sum_{i=1}^{d} \binom{l}{i} < \left(\frac{2B}{\varepsilon}\right)^{d} \left(\frac{el}{d}\right)^{d} \tag{28}$$

The Vapnik-Chervonenkis dimension (or VC-dimension) of $\boldsymbol{F}$ is the size of the largest shattered subset of X. Equivalently, VC-dimension of $\boldsymbol{F}$ is the largest value of l for which the growth function equals to 2^l. Then the covering number for EOT can be expressed by the domain of samples with grid size of $4W^2/\varepsilon^2$ within the VC-dimension. The covering number for EOT with VC-dimension bounds is given by

$$N_\infty(\varepsilon, \boldsymbol{F}, l) < 2\left(\frac{4lW^2}{\varepsilon^2}\right)^{d \log_2(2eWl/d\varepsilon)}. \tag{29}$$

We can now use the covering number of EOT with the Equation (14), but it must be noted here that $\varepsilon = \varepsilon'/2$ because of the nature of branching of the leaf node. Then the generalization equation for EOT with $d = (1 + 128V/\varepsilon)$ is given by

$$P_{Y,X}^{l}\left[I(\hat{f}) - I(f^*) > \epsilon\right] < 16\left(\frac{4lW^2}{\varepsilon^2}\right)^{d \log_2(2eWl/(d\varepsilon))} e^{-\varepsilon^2 l/2048}. \tag{30}$$

Generalization of AAKR

In AAKR, the empirical model is developed using memory vectors. A memory matrix contains the memory vectors and is represented by $\boldsymbol{X}$, where $X_{i,j}$ is the ith observation of the jth variable. For n_m memory vectors and p process variables, the memory matrix is given by,

$$\boldsymbol{X} = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,p} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n_m,1} & X_{n_m,2} & \cdots & X_{n_m,p} \end{bmatrix}$$

A query vector is the vector which is approximated by the AAKR and represented by $1 \times p$ matrix $\boldsymbol{x}$.

$$\boldsymbol{x} = [x_1 \quad x_2 \quad \cdots \quad x_p]$$

The first step is calculating the distance between the query vector and each of the memory vectors. A Euclidean distance is given by

$$d_i(X_i, x) = \sqrt{(X_{i,1} - x_1)^2 + (X_{i,2} - x_2)^2 + \cdots + (X_{i,p} - x_p)^2}. \tag{31}$$

The calculation is repeated for each of the n_m memory vectors for a single query vector. The distance matrix $\boldsymbol{d}$ has a dimension $n_m \times 1$ with the following form.

$$\boldsymbol{d} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n_m} \end{bmatrix}$$

The next step is calculating weights by evaluating the Gaussian Kernel with bandwidth h by

$$w = \frac{1}{\sqrt{2\pi h^2}} e^{-d^2/h^2}. \tag{32}$$

The weight matrix $\boldsymbol{w}$ with dimension $n_m \times 1$ with the following form:

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{n_m} \end{bmatrix}$$

Then, the final step is combining weights with memory vectors to make predictions:

$$\hat{x} = \frac{\sum_{i=1}^{n_m}(w_i X_i)}{\sum_{i=1}^{n_m} w_i}. \tag{33}$$

All the steps in AAKR can be written with a single function which approximates x to $\hat{x}$.

$$\widehat{x_j} = f(x_j) = \frac{\sum_{i=1}^{n_m}\left(\frac{1}{\sqrt{2\pi h^2}} e^{-\frac{1}{h^2}\left(\sqrt{\sum_{k=1}^{p}\left(X_{i,k}-x_k\right)^2}\right)^2} X_{i,j}\right)}{\sum_{i=1}^{n_m}\frac{1}{\sqrt{2\pi h^2}} e^{-\frac{1}{h^2}\left(\sqrt{\sum_{k=1}^{p}\left(X_{i,k}-x_k\right)^2}\right)^2}}. \tag{34}$$

For the ease of the calculations, it is possible to define a weight function σ by

$$\sigma_i(x) = e^{-\frac{1}{h^2}\|X_i - x\|^2}. \tag{35}$$

The single function of AAKR can be expressed in terms of weight function:

$$\widehat{x_j} = f(x_j) = \frac{\sum_{i=1}^{n_m}\left(\sigma_i(x)\, X_{i,j}\right)}{\sum_{i=1}^{n_m}\sigma_i(x)} \tag{36}$$

The bandwidth needs to be *a priori* fixed; otherwise, the weight function is not Lipschitz continuous. Derivatives of σ with respect to X and h approach to ∞ as $h \to 0$; therefore, there is no bound to function growth, and a generalization error cannot be established. For a fixed h, however, it is possible to derive generalization equations for AAKR. In this case, the parameter vector $w = (w_1, w_2, \dots, w_{n_m p})$ consisting of memory-vector parameters $X_{1,1}, \dots, X_{1,p}, \dots X_{n_m,1}, \dots, X_{n_m,p}$. The hypothesis space then can be written as $\mathcal{F}_{W,h} = \{f_{w,h} : w \in [0,1]^{n_m p}\}$ and an estimate function f is chosen from the hypothesis space.

In order to find the covering number of AAKR, we first need to calculate the Lipschitz constant. The weight function is bounded by 1 which means independent of X values, the weight function is always less than 1. This allow us to bound derivative of weight function with respect to parameters of X.

$$\frac{\partial \sigma_i}{\partial X_{i,j}} = \sigma_i(x)\frac{-2}{h^2}\left(X_{i,j} - x_j\right) \leq \frac{2}{h^2} \tag{37}$$

The next step is finding the derivative of estimate function with respect to parameters of X. For ease of calculations, estimate function can be written without the sum to take derivatives:

$$f_w = \frac{\sigma_i(x)\, X_{i,j} + C_1}{\sigma_i(x) + C_2}, \tag{38}$$

where the coefficients are $C_1 = \sum_{n\neq i}^{n_m} \sigma_n(x)\, X_{n,j}$ and $C_2 = \sum_{n\neq i}^{n_m} \sigma_n(x)$. The Lipschitz constant can be found by taking derivative of σ with respect to $X_{i,j}$ and using interval arithmetics:

$$\frac{\partial f_w}{\partial X_{i,j}} = \frac{\left(\frac{\partial \sigma_i(x)}{\partial X_{i,j}} X_{i,j} + \sigma_i(x)\right)(\sigma_i(x)+C_2) - \frac{\partial \sigma_i(x)}{\partial X_{i,j}}(\sigma_i(x)\, X_{i,j} + C_1)}{(\sigma_i(x)+C_2)^2} \leq \frac{4+h^2}{h^2 A^2 n_m}. \tag{39}$$

It must be noted here that, in theory, as d approaches ∞, the weight function goes to 0, which restrains the Lipschitz property. However, d is bounded by the memory vector; therefore, the weight function has a lower limit A and can be expressed as $\sigma \in [A, 1]$. We can find A by using the bound of d. Note that *Equation 29* gives a bound to d, and the bound can be expressed as $d \leq \sqrt{p}$. Then lower limit of weight function can be expressed as $\sigma_i(x) \geq A = e^{-\frac{p}{h^2}}$. Then the Lipschitz constant for AAKR can be calculated by

$$k_w \leq \frac{4+h^2}{h^2 n_m e^{-\frac{2p}{h^2}}}. \tag{40}$$

A uniform grid-cover size of $(1/\varepsilon)^{n_m p}$ satisfies the Lipschitz property; therefore, the covering number can be written as $N_\infty(\varepsilon, \mathcal{F}_W) \leq (k_w/\varepsilon)^{n_m p}$. Then the covering number for AAKR is given by

$$N_\infty(\varepsilon, \mathcal{F}_W) \leq \left(\frac{4+h^2}{\varepsilon h^2 n_m e^{-\frac{2p}{h^2}}}\right)^{n_m p}. \tag{41}$$

Implementing the covering number for AAKR gives us the generalization equation for AAKR:

$$P_X[I(f_{\hat{w}}) - I(f_{w^*}) > \varepsilon] \leq 8\left(\frac{128+32h^2}{\varepsilon h^2 n_m e^{-\frac{2p}{h^2}}}\right)^{n_m p} e^{-\varepsilon^2 l/512}. \tag{42}$$

Appendix B

AGR Thermocouple Data Analysis Using AAKR

This appendix describes the use AAKR analysis on measured data from the AGR fuel-irradiation test program conducted in the Advanced Test Reactor at Idaho National Laboratory. The AGR data set contains thermocouple (TC) measurement of graphite in six capsules. TCs work under very-high-temperature condition, which makes failure common. Each capsule underwent 12 irradiation cycles from February 2007 to July 2009.

The available data set contains measurements from five TCs in the experiment. Because these five measurements are highly correlated, AAKR can be applied to detect sensor drift. Due to the TCs' failures, only Capsule 6 has ≥3 TCs surviving with data recorded. Because AAKR requires at least 3 variables, only Capsule 6 (with its five surviving TCs) was analyzed by AAKR.

The AAKR average auto-sensitive is 0.4012 and cross sensitive is 0.3926. To reduce the missing alarms and false alarms, sequential probability ration test (SPRT) and alarm consolidation are applied to the residuals obtained from AAKR. SPRT was developed by Wald in 1947 as a statistical test of whether a measurement is more likely from a normal operation, H0, or from an abnormal operation (whether due to drift or fault), H1. The advantage of SPRT is it requires few numbers of samples to determine an anomaly.

The fault hypotheses from SPRT are consolidated by removing spurious alarms with a percentage check technique. This technique determines that the signal is a fault when two of five successive observations exceed the threshold. Figure B-1, through Figure B-5 show the fault hypothesis results for TC 1–5, respectively. The blank region background is the region with missing values where no analysis is conducted. X axis shows the observation index, and Y axis shows the fault hypothesis, where 0 indicates normal operation, and 1 indicates abnormal mode. The yellow and green backgrounds show different Advanced Test Reactor (ATR) cycles, the details of which are given as Table B-1:

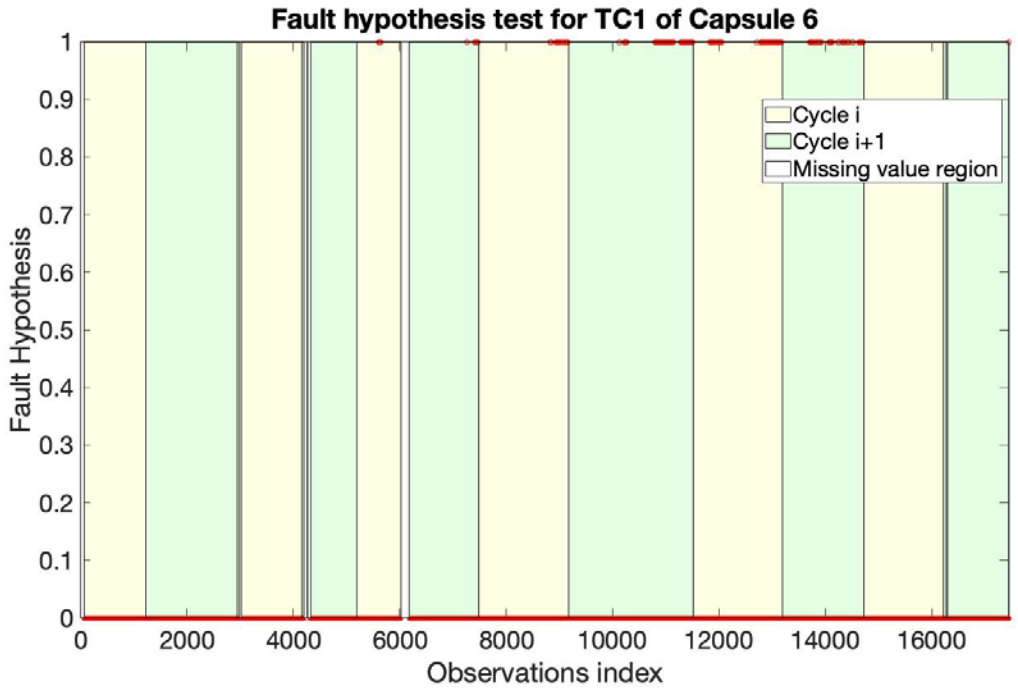


Figure B-1. Fault hypothesis for TC1 of Capsule 6.

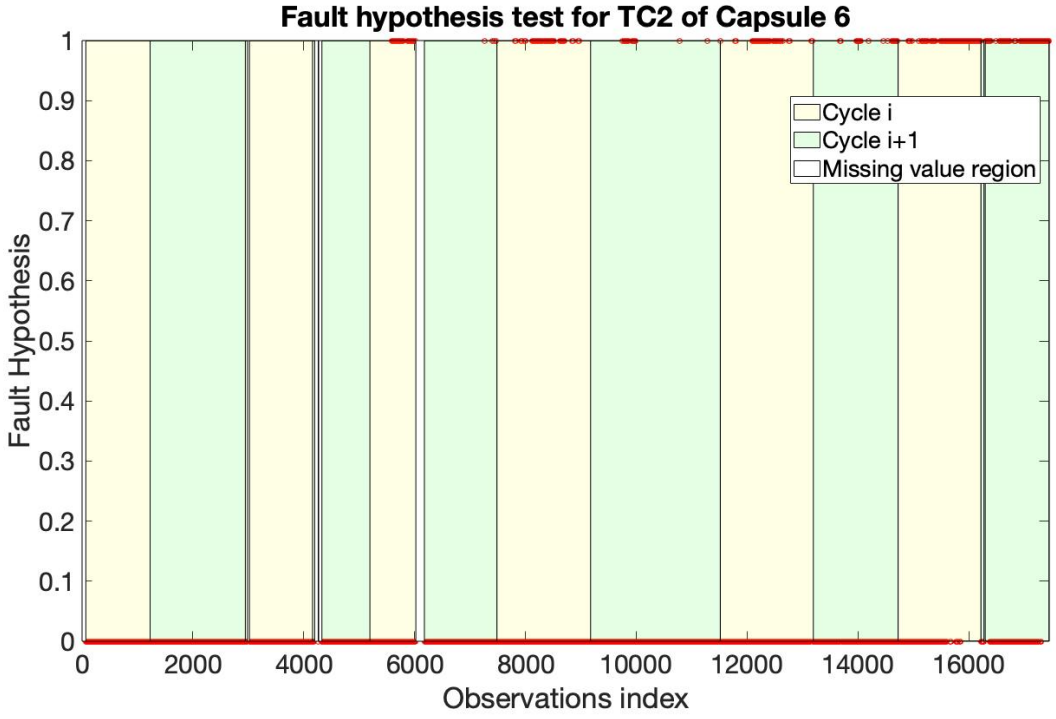


Figure B-2. Fault hypothesis for TC2 of Capsule 6.

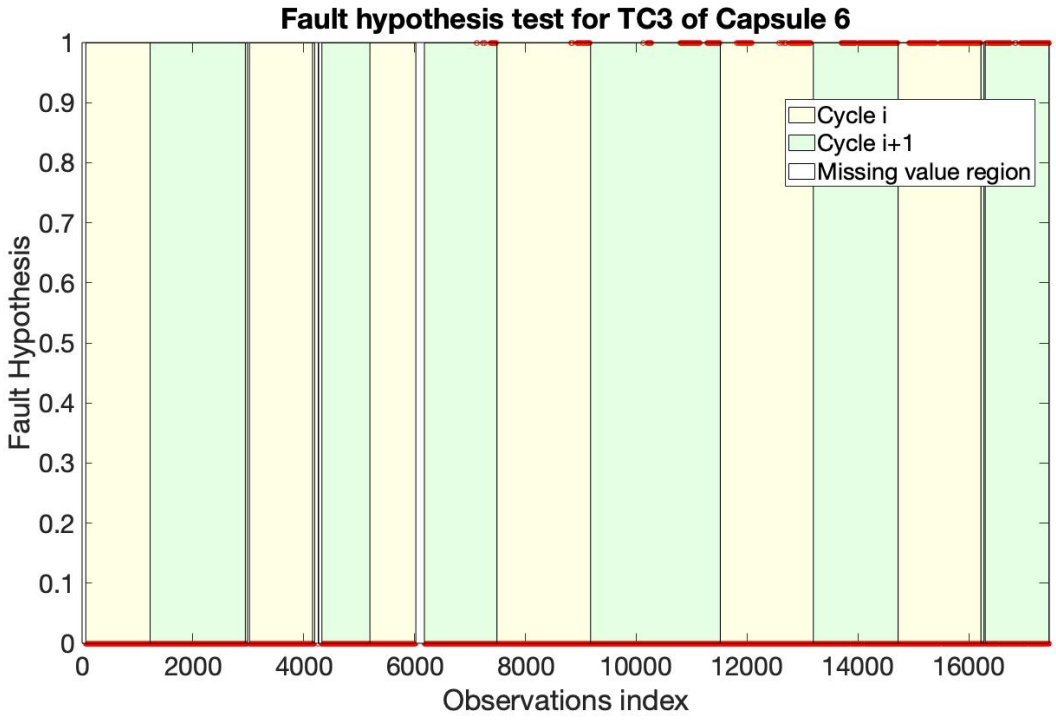


Figure B-3. Fault hypothesis for TC3 of Capsule 6.

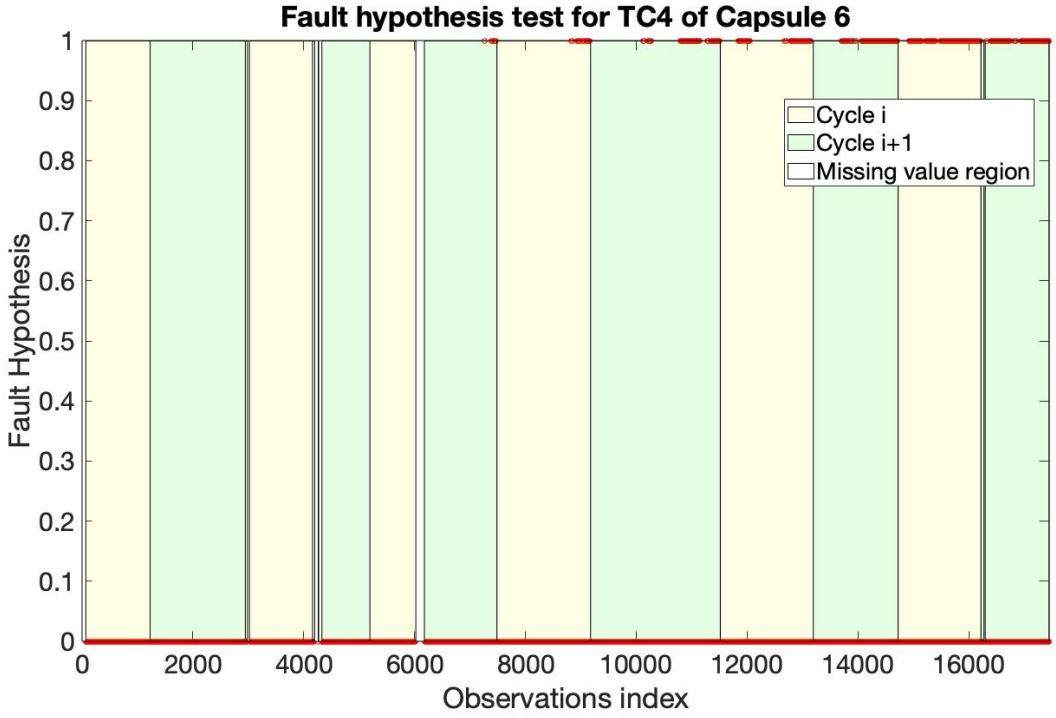


Figure B-4. Fault hypothesis for TC4 of Capsule 6.

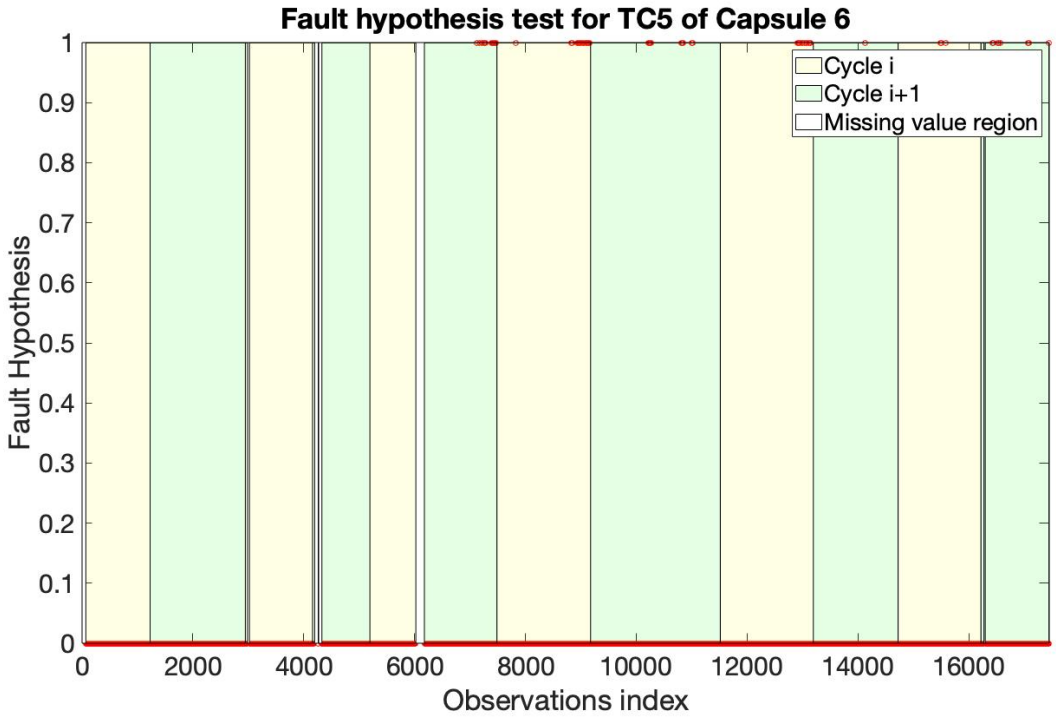


Figure B-5. Fault hypothesis for TC5 of Capsule 6.

Table B-1. ATR cycle details for TC 1–5.

ATR Cycle	Start Date
139A	10 Feb 2007
139B	21 Apr 2007
140A	29 Sep 2007
140B	1 Dec 2007
141A	26 Jan 2008
142A	8 Mar 2008
142B	21 Jun 2008
143A	30 Aug 2008
143B	6 Dec 2008
144A	20 Feb 2009
144B	25 Apr 2009
145A	4 Jul 2009

The results suggest that the drift/degradation starts from around the middle of cycle 141 A for TC1 and TC2, and late in cycle 142A for TC3-5.