ORNL/TM-2020/1697

Development of Prognostic Models Using Plant Asset Data

Pradeep Ramuhalli
Cody Walker
Vivek Agarwal
Nancy J. Lybeck

September 2020

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE US DEPARTMENT OF ENERGY

ORNL/TM-2020/1697

Reactor and Nuclear Systems Division

DEVELOPMENT OF PROGNOSTIC MODELS USING PLANT ASSET DATA

Pradeep Ramuhalli
Oak Ridge National Laboratory

Cody Walker
Vivek Agarwal
Nancy J. Lybeck
Idaho National Laboratory

Date Published: September 2020

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831-6283
managed by
UT-BATTELLE, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

FIGURES

TABLE

ACRONYMS

AI	artificial intelligence
BOP	balance of plant
BWR	boiling-water reactor
CI	confidence interval
FWCS	feedwater and condensate system
LSTM	long short-term memory
ML	machine learning
NAR	nonlinear autoregressive
NDE	nondestructive evaluation
PCA	principal component analysis
RMSE	root mean square error
RNN	recurrent neural network
RUL	remaining useful life
SVR	support vector regression

ABSTRACT

The recent growth of machine learning and artificial intelligence technologies provides opportunities for leveraging data-driven algorithms to address the problems of diagnostics and prognostics in the nuclear power industry. The use of machine learning and other statistical methods as prognostic models is of particular interest in the nuclear industry to accurately predict future equipment or plant state given a set of measurements. Such predictive capability will enable predictive assessment of component condition and remaining life and allow for condition-based predictive maintenance. The resulting optimization of maintenance scheduling and reduction in unnecessary maintenance activities will lower overall maintenance costs and improve the economics of nuclear power. This report discusses the various aspects of data processing and model development that are likely to influence the performance of prognostic models. Data from a boiling-water reactor was used to evaluate several prognostic models to identify key considerations for developing such models to predict data-driven plant state and equipment degradation condition. Preliminary results indicate the need for data sets that are relevant to the problem at hand and contain signatures that may be correlated to the prediction problem. Assuming such data exist, development of prognostic models using data-driven methods requires an understanding of the various sources of influence on the prediction accuracy (such as the model architecture, data preprocessing approaches, and potentially external factors influencing the equipment or plant system under assessment). Ongoing research is evaluating these factors in greater detail and examining techniques for calculating prediction uncertainty bounds.

1. INTRODUCTION

Nuclear plant sites collect and store large volumes of data from equipment and systems at different temporal and spatial scales. These data typically include plant process parameters, maintenance records, technical logs, online monitoring data, and equipment failure data. The data may be collected using handheld instrumentation at irregular time intervals, or it may be streamed and archived on the plant computer. Structured methods and associated tools must be developed to analyze these data to provide actionable information on the current and future state of critical plant equipment. If such diagnostic and prognostic (predictive) insights into the state of the equipment were made available, then they could be used to drive maintenance decisions to improve equipment availability and reliability. However, the potentially unstructured nature of the data challenges the ability to develop scalable and reliable methods for data analysis.

The recent advancements in machine learning (ML) and artificial intelligence (AI) technologies provide opportunities for leveraging data-driven algorithms to address the problems of diagnostics and prognostics in the nuclear power industry. Of particular interest is the development of generalizable prognostic methods that can accurately predict future equipment state given a set of measurements correlated to the current state of the equipment. Embedded in this problem is the ability to assess the current state of the equipment based on these measurements, as well as the need to develop a suitable model or models to estimate the change in equipment condition over a prescribed time horizon. The resulting estimate of the equipment's remaining useful life (RUL) can improve overall plant performance and reduce maintenance costs through optimization of maintenance activities.

While many approaches to prognostics have been described in the literature, there remains a need for objectively assessing prognostic models, especially those using ML, based on data from operating plants. An associated need is the development of a formal methodology for using and integrating these techniques into the nuclear industry maintenance practices.

1.1 SCOPE OF THIS REPORT

This report examines the challenges associated with developing and assessing prognostic models, especially those using common ML techniques, as a step towards developing the formal methodology for stakeholder use and integration.

The overall goal of this project is to develop and demonstrate predictive maintenance using online monitoring of critical systems in the balance of plant (BOP) of nuclear power plants (NPP). This report focuses on prognostic models, which are one of the key technologies for achieving this goal. The specific objectives of the work described in this report are to apply and evaluate ML algorithms for prognosis of equipment health and performance using relevant metrics. As necessary, this research defines the metrics to be used in the evaluation that are relevant to nuclear plant maintenance requirements.

This report is the most recent in a series of documents examining online monitoring, application of advanced sensor technologies, and development of diagnostic and prognostic models for critical BOP equipment in NPP [1-4]. Previous efforts assessed the application of wireless technologies in NPPs which included evaluating potential wireless communication networks [4]; technical and economic evaluation of the wireless network that can support different communication protocols [2, 3]; and an evaluation of vibration sensors that can enable wireless data transfer in nuclear plants [1].

While vibration data are useful for equipment condition monitoring and are used in operating plants for assessing condition of BOP equipment, limitations in the data sets available to the project team required the research to focus on using plant operational data for assessing prognostic models. Therefore, this analysis used data from the condensate and feedwater systems in a boiling water reactor to build prognostics models.

1.2 ORGANIZATION OF THIS REPORT

Section 2 discusses prognostics models—including the problem formulation, classes of models, and the selected models for this study—and it provides an overview of data preprocessing needs. Section 3 discusses preliminary results and associated findings, and Section 4 summarizes the report and outlines ongoing and planned future work.

2. PROGNOSTIC MODELS

2.1 BACKGROUND

The report by Meyer et al. [5] discusses the requirements for a prognostic health management system for advanced small modular reactors and documents the needs to address technical gaps identified through a gap analysis effort. The requirements documented therein were derived from a number of application-specific drivers such as reactor design, operations and maintenance concepts, materials, environmental conditions, and other factors; these requirements were generally focused on passive components. However, a more recent review appears to indicate that many of the research needs identified in Meyer's work are applicable to other reactor concepts and other classes of components [6].

Previous studies have identified prognostic models as a key technical need that can integrate data from multiple sensors for robust prediction of remaining life, uncertainty quantification for the prognostic result, and lifecycle prognostics models. Most of the analytics research that has been performed to develop diagnostic and prognostic estimates in the nuclear industry has used data from a single sensor type. However, correlation techniques have been developed for diagnostic and prognostic estimates based on interaction between time-domain and frequency-domain features [7]. In recent years, research has been

performed to develop diagnostic and prognostic estimates by integrating sensor and maintenance data [8, 9]. In addition, analytics research to date in the nuclear industry has not taken into account relevant information about the asset such as age, maintenance history, mode of operation, level of maintenance performed, and whether the performed maintenance activities returned the asset to a like-new state or to a more intermediate state of health.

Prognostic or predictive models, as the names indicate, are intended to predict the value of a variable given one or more input parameters. While the terminology may be generally applied to any model that relates input and output variables, in this report, the definition of prognostic models is restricted to models that (1) use time-series data as inputs, and (2) predict the value of the output variable at some time in the future. In all cases, data from multiple sensors are assumed to be available as model inputs, allowing the model to implicitly integrate data from multiple sensors for robust predictions.

The prediction time horizon in such models is usually fixed and implicit (i.e., not an explicit input to the model). The input time-series may be the raw measurements that have not been processed in any manner (to enhance the signal-to-noise ratio, for example), or they may be features extracted from the measurements, such as the principal components of the measurements.

Development of a prognostic model begins with defining the output measure to be predicted. This could be the generated output power of a nuclear power plant or the length of a crack in a structural material. The next step is selection of the prognostic parameters, which are relevant features from the data that are somehow related to or useful for predicting the output measure. The next step is the selection of the model form, after which the model parameter estimation, testing, and validation can proceed. In general, the selection of a model is informed by the type of data available: labelled vs. unlabeled data, large vs. small sample size, type of correlation between the prognostic parameters, etc.

2.2 PROBLEM FORMULATION

This research was initiated to develop prognostic models that are useful for RUL prediction for a component of a subsystem. The RUL is defined as the time remaining till the estimated (predicted) condition of the plant asset becomes functionally unacceptable, with the condition estimate computed by the prognostic model based on data from the asset. The problem of RUL prediction assumes that the component or subsystem of interest degrades over the course of operations. The time remaining before the component or subsystem is unable to meet its functional requirements must be determined. One specific approach to formulating the problem is as follows.

Let $\mathbf{x}(t) = [x_1(t), x_2(t), \dots x_i(t), \dots, x_N(t)]$ be the vector of measurements from N sensors. Furthermore, let $y_j(t) = f_j(\mathbf{x}(t)), j = 1,2, \dots, M$ be a set of $M \leq N$ prognostic indicators computed from the N measurements using a known set of functions $f_j(\cdot)$. Assume that a prognostic model $g(\cdot, \Theta)$ parametrized by Θ is available such that the sequence of prognostic indicators $\mathbf{y}(t) = [y_1(t), y_2(t), \dots y_M(t)]$ for $0 \leq t \leq t_k$ may be used to predict the value of a health index $H(t)$ for $t > t_k$:

$$\widehat{H}(t) = g([\mathbf{y}(t_0), \mathbf{y}(t_1), \dots \mathbf{y}(t_k)], \Theta) \quad (1)$$

where $\widehat{H}(t)$ indicates the estimate of $H(t)$. The RUL is then simply the time interval between t_k and the time at which the health index is no longer acceptable.

This simple formulation, while acceptable for many problems, may be unacceptable for nuclear plant equipment health prognostics, as it does not include a formal mechanism for quantifying the uncertainty

in the prediction. Several other formulations that are generalizations of Eq. (1) are available to estimate the prediction uncertainty [10], but the formulation in Eq. (1) is sufficient for the present assessment.

Several assumptions are embedded in Eq. (1), including the fact that RUL prediction will require the ability to compute prognostic indicators from the measurements (data) that are sensitive to the amount and type of degradation in the component or subsystem health.

However, it was determined that the available data from an operating plant (see Section 3.1 for details) did not have any examples of equipment degradation or failure that were encapsulated by the measurements. As a result, the decision was made to focus initially on the simpler problem of predicting plant conditions (i.e., one or more measured plant process variables) at future time instants based on past measurements of plant process parameters. Note that prediction of one or more measured quantities from plant process parameters can also be formulated using the formulation highlighted in Eq. (1), with $H(t)$ replaced by the relevant measurements $x_i(t)$ or the prognostic indicators $y_j(t)$ in Eq. (1).

Within the context of predicting plant conditions, two specific sub-problems may be identified. The first of these is the prediction of plant output (generated power) at a given time from measurements of plant process parameters such as feedwater flow and temperature and equipment operation parameters such as motor current and pump differential pressure at that time or at previous time instants. The second sub-problem is one of predicting the plant process parameters or equipment operation parameters given the data from those parameters at previous time instants.

Each of the two plant condition prediction scenarios may be formulated using a similar setup as that used in Eq. (1) and are used in the evaluations in this study to elicit insights into model performance. However, the second sub-problem (plant/equipment parameter prediction) has similar characteristics to degradation prognostics in that the prognostic model represents the change in the quantity of interest over time.

2.3 PROGNOSTIC MODELS - CATEGORIES

In general, prognostic models can be divided into three subcategories: physics-based models, data-driven models, and hybrid models.

Physics-based prognostic models attempt to predict the future value of the output variable using measured data in combination with a physics model. Such a model might be, for instance, based on a set of partial differential equations or ordinary differential equations that encode the first-principles relationship between the inputs and outputs [11, 12]. An example of a physics-based prognostic model is a crack growth model such as Paris' law to predict fatigue crack length as a function of time to the number of fatigue cycles and the stress intensity factor [13, 14].

Physics-based models typically produce high-accuracy results and require less data for tuning when the mechanism is well known [15]. However, physics-based models are usually computationally expensive, especially when applied to system-level prognostic problems or when multiple degradation modes need to be represented. A classic example of this is degradation prognostics, in which the problem is one of predicting the remaining life of a system that can fail from one of multiple degradation modes. Without *a priori* information on the specific degradation mode, a physics-based prognostic model can be difficult and expensive to formulate and solve. Even for a simple model, many assumptions and estimations are made during the creation of the model, leading to suspicions about physical model adequacy [16].

Data-driven prognostic models estimate the value of the output variable through observed data only and do not incorporate first-principles information about the relationship between the input and the output.

Data-driven methods rely on trends within the data to construct mathematical models to predict future states of the system.

The performance of data-driven methods, including statistical and ML methods, are dependent on the amount and quality of available data that may be used to infer the model parameters [16]. The key difference between statistical and ML methods is the focus of ML methods on prediction (using models to estimate the output for new input data) instead of inferencing (using models to understand the data generation process). While models using machine learning are valuable for learning hidden higher order relationships from data, the performance of these methods is sensitive to the model's structure (number and size of layers, connectivity between layers), learning algorithm (e.g., Levenberg-Marquardt, Bayesian regularization, and gradient-based methods [17]) and the initial values of the model parameters (weights).

Data-driven methods are beneficial because they can be deployed on large, complex systems; every mechanism or interaction does not need to be known to produce acceptable results. Data-driven methods are easy to implement with a low cost of creation [18]. Once the data have been collected, models can be developed extremely quickly as compared to physics-based models. But data-driven models have issues as well. They require large amounts of data covering a wide range of conditions. The outcomes produced usually have a high confidence only in the domain spanned by the training data. Any predictions made outside this domain will lead to extrapolation and potentially non-physical results [18]. For their use in the creation of degradation models, run-to-failure data are required. This type of data may not exist for high-priority or new systems. Determination of an appropriate failure threshold can also be difficult, even when run-to-failure data exist.

Physics-based and data-driven methods each have their advantages and disadvantages, which hybrid methods attempt to combine for the best results. This area of research has not matured, but it is growing [13, 19]. Many of these models are application specific, with one model predicting the health state while the other predicts the RUL. Another methodology attempted to incorporate both model types into the RUL forecasting stage. Eker et al. [20] used a physics-based model to make near-term prediction and then used data-driven methods to compute long-term forecasting.

Appropriate model selection depends on knowledge of the system behavior and available data. While physics-based models may be available in some instances, the vast majority of applications tend to focus on data-driven methods because of the complexity of the physics models and the relative ease of access to data from the system of interest. Within the category of data-driven models, factors dictating model selection include:

- Objective of the prognostics (anomaly detection vs. predicting health state), which impacts the problem formulation and the associated model selection (classification vs prediction)
- Amount of available data
- Labelled data availability. If labelled data that associate the data with the system state or condition are available, then supervised learning methods are an option and may dictate the type of model chosen. Otherwise, unsupervised methods are preferred and correspondingly limit the types of models
- Type of data (time-series vs. state space, run-to-failure data, etc.)
- Statistical characteristics of the system and available data (for example, stationarity of the system), which will affect the data processing stages and the model selection

2.4 PROGNOSTIC MODELS SELECTION

As indicated above, there are several options for prognostic model selection, and the specific choice is dependent on the objective of the prognostics. For the problem of RUL estimation, for instance, model

selection is largely based on knowledge of the system and potential modes of degradation. This approach will determine whether a physics-based, data-driven, or hybrid model is the best option. If the objective is to predict future plant conditions (i.e., future values of select measurements from a plant), then model selection will depend on the level of available knowledge and data on the system behavior.

While physics-based models are a possibility for predicting future values of select measurements, model complexity may be prohibitive when dealing with a complex system such as the feedwater and condensate system. A data-driven approach reduces the complexity of the model development and may be useful in identifying anomalous behavior in the system performance [16] that may be an early indicator of degradation in critical components. Based on the simplicity of developing and applying data-driven models and the potential for applying recent advances in machine learning for this purpose, three data-driven models were selected in this study: long short-term memory (LSTM) neural network, nonlinear autoregressive (NAR) network, and support vector regression. These three models are distinguished by their ability to work with time-series data, are relatively simple to develop and apply, and appear to be able to represent predictive relationships between time-series over different time horizons [16]. Note that other modeling options are also available [21-23] and may be appropriate for other prognostic applications.

The LSTM network [24] is a type of recurrent neural network (RNN) that includes unique memory cells specifically developed to learn and encode long-term relationships between inputs and outputs. This is in contrast to classical RNNs that, while utilizing temporal information to develop the relationship between previous inputs to inform the current output, are challenged with identifying and encoding longer term relationships present in the data [25]. In the LSTM, the memory cell stores the state within the network that interacts with the current input. The stored state is updated as new inputs are available; a “forget” gate determines which information to forget from the previous memory cell state. The output from the forget gate is then transferred as a hidden output to inform the memory cell during the next time step. A number of activation functions may be used as part of the LSTM cell as part of the internal state computation and update [26]. A detailed guide to LSTM is provided by Greff et al. [27].

A NAR neural network is a dynamic recurrent network. A NAR network is a more traditional RNN that consists of a single layer, while the LSTM is a more complex RNN consisting of multiple layers, each serving a different function. Dogan used both NAR and LSTM models to predict network traffic flow and showed that the LSTM outperformed the NAR model under most circumstances [25]. This model makes predictions based on historical values and can be used to forecast nonlinear time series [28]. The NAR model is trained in an open-loop style using backpropagation to update the values within the hidden layer. This model is trained to make one-step-ahead predictions, but the loop can be closed to make further predictions. To close the loop, the NAR network is given initial values, and the output is directly piped back into the inputs. The success of the NAR model depends on the architecture of the model and the selection of the inputs. Various combinations of hidden nodes, delays, and inputs should be tested to ensure the best fit. Increasing the number of hidden neurons can increase the model’s ability to fit the training data. Too many hidden neurons, however, will increase training time and can overfit the training data, leading to poor generalization.

Support vector regression (SVR) is fundamentally a kernel-based regression model based on the core concept of support vectors; this model can be used for classification or regression [29]. SVR has two primary components: a kernel function and an optimization routine. SVR first transforms the input data, , not linearly separable in the input data space, into a higher dimension feature space using a kernel function [30]. The optimization routine seeks to minimize generalization error, which is the sum of the training error and the confidence level. The solution depends only on a subset of training data points called the *support vectors* [31]. SVR is notably different from other regression techniques because

training data within a tolerance do not penalize the loss function being optimized. SVR is noted for its accuracy, regularization, and generalization abilities [18].

2.5 DATA PREPROCESSING NEEDS FOR PROGNOSTIC MODELING

2.5.1 Prognostic Indicators

A prognostic indicator can be anything used to measure or infer the health of a component or system of interest. Passive components such as the containment or welds rely on a variety of nondestructive evaluation (NDE) techniques to measure defects, thermal fatigue, creep, etc. Active components such as motors, turbines, and heaters can be measured directly through vibration, acoustics, temperature, pressures, current, flow, etc. Raw instrument signals can also be combined to produce prognostic indicators. The optimal prognostic indicator depends on the system, the outcome being predicted, and the available measurement signals. For example, useful signals for an induction motor could include current, vibration, and bearing temperatures [32].

2.5.2 Data Preprocessing

Rarely are the raw measurement signals the optimal choice for a prognostic indicator. Raw data should be processed and cleaned to improve prognostic performance. Raw data can be subject to outliers, sensor and process noise, missing information, and more. These types of errors can be filtered and replaced. Data processing includes feature identification, normalization, and selection [33].

Feature selection is the process of choosing signals that aid in prognostic modeling while eliminating irrelevant or redundant signals. Some measurement signals such as vibration are very rich in information. In these cases, it advantageous to select a feature from the data rather than using the raw signal itself. Features such as root mean square, kurtosis, or an amplitude of a specific frequency peak may be more related to the output being predicted. The optimal feature is dependent on the system and the output being predicted. Multiple features should be tested when able. Principal component analysis (PCA) can be used to reduce the dimensionality of the feature space while retaining the variance contained within the data. Each of the principal components is orthogonal to one another, meaning they are uncorrelated [34]. By selecting principal components that cumulatively explain 95% of the variability in the data, the majority of the information is retained while the total number of inputs required for the model is reduced.

Normalization is typically used to ensure that the models are sensitive to variations in the features of interest and not to the absolute values of the features. Normalization is usually performed in one of two ways. The first approach, which is referred to as *standardization*, scales the data to have a specific mean and variance. The second approach scales the data so that the data lie in a fixed range. Data are typically normalized prior to being used in neural networks, but this is rarely done in statistical models [33]. The LSTM can be sensitive to input scaling, so inputs should have a mean of 0 and a variance of 1.

2.5.3 Evaluation Metrics

Metrics have been developed to evaluate the potential performance of prognostic indicators. These metrics include *monotonicity*, *trendability*, and *prognosability* [35]. *Monotonicity* captures the overall positive or negative trend of the feature. When working with mechanical equipment, the major assumption is that degradation is cumulative as long as there is no repair or maintenance. A strongly monotonic signal can be useful for predicting continuous degradation. *Trendability* represents the underlying shape that the feature's population exhibits and how well a function could fit that population. *Prognosability* measures how well the critical failure values are clustered, as well as the variance between

the initial and failure values. Tightly clustered failure values indicate that the failure threshold can be confidently defined. A large distance between initial and failure values can aid in the differentiation between a healthy or failing component.

Evaluation metrics judge models based on their predictions. Each metric compares a different aspect of the prediction, so it is important to be consistent in the metric choice when comparing a model to others. The choice of which metric to use depends on how much information is available other than the model's prediction. If the prediction is offline and the ground truth is known, then root mean square error (RMSE) or confidence interval (CI) should be used. The RMSE is a commonly used procedure to measure the average prediction error between the model prediction and ground truth. A smaller RMSE means a closer average prediction. Alternate metrics quantify prediction accuracy in terms of the maximum prediction error (Maximum Absolute Error or MAE) and residual standard error (RSE, which is a variant of RMSE adjusted for the number of prognostic indicators). A CI is a measure defining how often the RUL prediction was within a specified range around the actual RUL at the time of prediction. A narrower CI means that the RUL prediction is both accurate and concentrated [36]. Lei [36] also provides a detailed review of several other evaluation metrics, including alpha-lambda accuracy, prediction horizon, predictability, and others. It is important to note that these metrics may be applied for comparing models on the basis of their performance on a test data set. However, good accuracy (for instance, low RMSE) on a training or test data set is not necessarily indicative of good generalization performance (prediction performance on data not previously seen by the model). While such generalization performance is difficult to quantify, generalization error bounds may be computed in some cases [37, 38].

2.6 LIMITATIONS

It is important to note that none of the models described above have the inherent capability to estimate the uncertainty in the model predictions without substantial modifications. Such modifications have typically used Bayesian or Monte Carlo methods [10] to quantify the model prediction uncertainty. At this writing, the focus of this effort has been to assess prognostic model accuracy and to define the implications of varying the key parameters associated with the different models. Assessment of uncertainty quantification methods for prognosis will be addressed in the future.

While prognostic model development is relatively straightforward if data are available, a challenge arises if new data become available and are expected to increase the accuracy and confidence in the model predictions if included in the model development process. Several options exist for model updating based on new data stream availability; these options vary from simple (retrain the entire model) to complex approaches that use evidence-based techniques to update the model parameters. In the work described here, the focus is on categorizing the data available and predicting future plant conditions using available data only. As new sensors and data streams become available, integration of data into prognostic models will be evaluated, with the leading candidate for this being a Bayesian approach.

3. PRELIMINARY RESULTS

This section provides an overview of the results achieved when applying the models described above for prognostics of plant state using data from an operational plant. As described above, the primary objective of this analysis is to assess the capabilities and limitations of these data-driven prognostic models relative to time-series prediction; that is, predicting future values of a time series given one or more time-series measurements from the past. This assessment requires the ability to quantify the necessary data handling and preprocessing needs, and the ability to determine the computational complexity associated with developing the models. Furthermore, the relative accuracy of the different models is important. These requirements are discussed in this section.

3.1 DATA DESCRIPTION

The data used in this work to evaluate prognostic models was from the feedwater and condensate system (FWCS) of a BWR. The primary task of the FWCS is to supply clean, demineralized water to the reactor vessel at the correct temperature, pressure, and rate. The data set contains process data, as well as condition-based and preventive maintenance work orders for various components throughout this system.

Available process data include variables such as the generator's gross load and the average feedwater flows and data on temperatures and pressures at the condensers, feedwater pumps, condensate and condensate boost pumps, and motors for multiple pump trains. The data from the plant historian were downsampled to hourly intervals and provided by the plant. The available data set is for a five year period covering multiple complete refueling cycles: steady-state operation, ramp-downs (shutdown for refueling), and several derates of varying levels. The available data were unlabeled, so there was no indication of whether any portion of the data corresponds to equipment failure. Furthermore, the data did not have sufficient information to determine the cause of each derate. Discussions with the plant engineers indicated that most of the derates were caused by factors other than the FWCS (i.e., no FWCS component failures were seen during the time period corresponding to the data). Additional information, in the form of reports summarizing the plant performance and any known issues, were also provided to the project team. The timing of this information delivery made it difficult to analyze and correlate with the process variable data; therefore, this analysis will be included in subsequent project reports.

For this report, the objective of the model was to predict plant conditions at one hour, one day (24 hours), and one week (168 hours) into the future. Each of the three model types was used as a way of comparing ML prognostic models (LSTM and NAR) against a statistical prognostic model (SVR). The available time series data from the process variables was cleaned (see Section 4.2), and the data from steady-state operations were separated from data corresponding to ramp-downs and derates. The available steady-state data were further partitioned into two groups: a training data set containing the first 80% of the steady-state data, and a test data set containing the remaining 20%. The models were each built using the training data and were evaluated against the test data. Model performance was measured using prediction accuracy and computational complexity associated with deriving the model from the training data.

It is worth noting that that the steady-state process variable values had differences (average plant output power differed by about 0.5% - 1%) from cycle to cycle. Preliminary analyses using the data from fuel cycles with similar plant power outputs to develop (train) the models demonstrated somewhat poor prognostic performance on the test data (other cycles), exhibiting a persistent bias in the predictions. This behavior is not inconsistent with the performance of these types of data-driven models, as their generalization errors tend to increase when based on data that are dissimilar to those used in the training. The 80/20 split in the data was chosen as a result to ensure sufficient diversity in the training data set and to eliminate an obvious source of bias in the predicted results. The 80/20 split included measurements from all fueling cycles.

Although it was not the primary aim of the research, a limited analysis used all the data (steady-state and derates/ramp-down) in an attempt to assess the capability of these types of models to represent nonstationary behavior. Again, the available data were partitioned into training and test data using an 80/20 split. Results from this limited analysis are discussed later in this section.

3.2 DATA CLEANING AND PROCESSING

Because of the varying states of operation, the data set was divided into four categories: steady state, refueling, derate, and trip. The steady state covered all instances when the reactor was above 90% of nominal full power. Although the power does fluctuate within this category, it is still covered within the

broadly labelled steady state. Refueling covers the period of time from the initial ramp-down through the refueling outage, with the ramp-down and subsequent startup representing non-steady–state operation. Derates contain all observations in which the reactor is operating between 5 and 90% nominal full power. Derates can occur due to operational, reliability, or environmental issues. The final category is trip, or scram, which is the emergency shutdown of the nuclear reactor. Trips are similar to derates in appearance, but instead of the gross load being reduced, the reactor is shutdown.

The data were cleaned before being processed in models. The data cleaning process was intended to address instances of missing data and outliers and to separate out the data from steady-state and non-steady–state conditions. Missing information was primarily noted for instances during which the component was offline. If the component was online but the data were not recorded due to an error in the sensor or data archiving process, then the missing values were interpolated using neighboring values. If the missing value corresponded to an inactive loop, then the component may have been taken offline, so the value was left as is. Daylight saving time was a minor inconvenience, as periods of time were "skipped" once per year. Time stamps were not duplicated when daylight saving time ended towards the end of the year. These time steps were filled as previous values.

Potential outliers were selected based on values being at least four standard deviations away from the average, as shown in Figure 1. For many of the variables in the steady-state portion of the runs, data are heavily skewed in one direction or the other. When searching for potential outliers in the BOP average flow, roughly 1% of the data was beyond three standard deviations. After expanding the threshold to four standard deviations, the number of outliers was reduced to 0.3% of the total dataset, as seen below. Once an outlier was found, this value was corrected using a median filter. Median filtering uses a sliding window approach to replace values within the median of the window. The window size is important during this process, and in this case, a window size of 51 points (slightly over 2 days' worth of data) was selected. For the purpose of anonymizing, the axes of the Figure 1 is normalized.

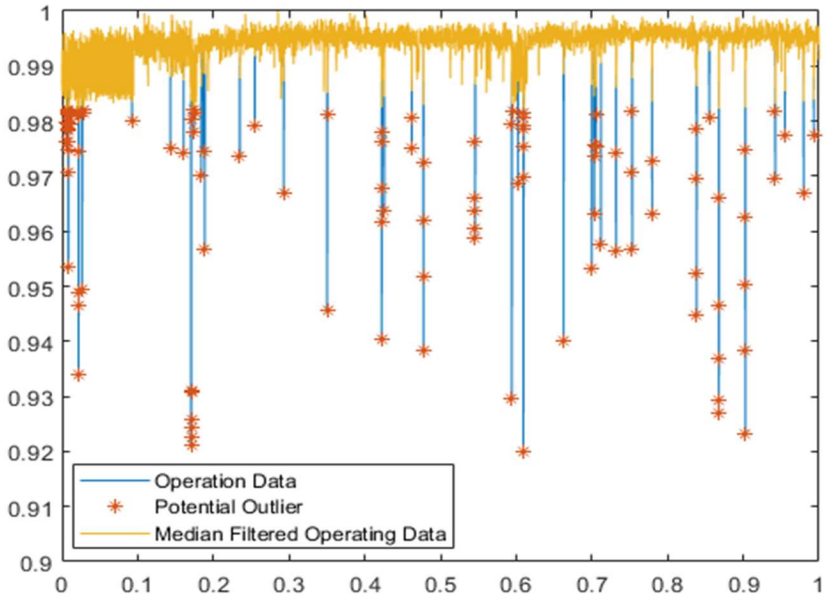


Figure 1. Example of outliers identified in the average total feedwater flow measurement. The data range has been normalized so that the maximum value of the variable is 1, and the normalized horizontal axis represents the fraction of plant operation time.

Temperature data can contain seasonal variations and long trains of outliers, as seen in Figure 2). Median filtering was used to address these problems. Median filtering using a larger window size of 700 was chosen to account for the seasonal variations.

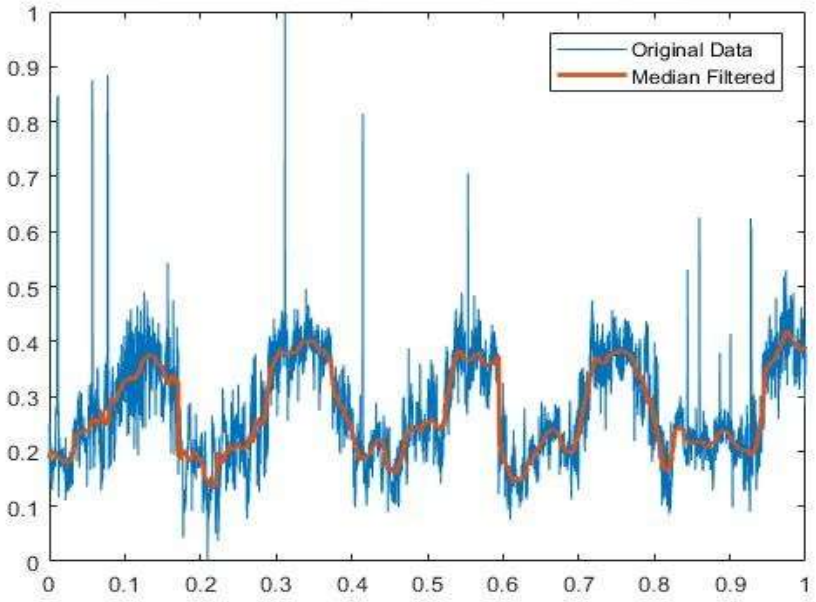


Figure 2. Example of seasonal trends or other periodicities observed in temperature measurements (feedwater temperature). Again, the temperature is scaled to be between 0 and 1, and the horizontal axis represents the fraction of plant operation time.

Subsequent to the data cleaning, the measurements were normalized (standardized) to zero mean and unit standard deviation (data distribution scaled to a standard normal distribution).

3.3 PROGNOSTIC MODELING PERFORMANCE

Each of the three models was used to estimate the prediction performance based on a subset of process variable measurements corresponding to the total feedwater flow and temperature measurements. To ensure a reasonable comparison between the modeling approaches, a subset of process variable measurements (feedwater flow, feedwater temperature, and feedpump/condensate boost pump operating characteristics) was used to build the models. This subset of data showed a high correlation with the generated load from the plant and was directly related to the equipment of interest for degradation detection. Model development and performance evaluation were performed using one of two methods. In the first method, the normalized data were used directly to build and evaluate the models. In the second method, PCA [34] was used to identify the largest principal components from the data. These constitute the components with the greatest explanatory power, with the remaining principal components more likely to be related to noise in the data. These data were used in the model development and evaluation. It is expected that the use of PCA will reduce the model complexity by reducing the size of the models and will enhance the robustness of the models by focusing on features that are more likely to be related to the system states of interest. A further benefit of the PCA approach is the de-correlation of the measurements by projecting the data along orthogonal principal axes; this is expected to improve the prediction accuracy.

3.3.1 Prognostic Model Results

The prognostic model performance, especially for the ML models, is expected to be sensitive to the model structure, parameters, and data used. To verify this, the LSTM and NAR models were trained with differing numbers of hidden layer neurons; performance was tracked through the time required to train the models (on a dual-core CPU system) and the root mean square error (RMSE) in the prediction. The RMSE values were estimated using the test data set only. The SVR models were used as a baseline for comparison of the other neural network-based models. In most cases, the models were applied to predict future values of one or more of the measurements. In a few instances, however, the models were applied to predict the generated output of the plant from the measurements. The input data streams were also varied, with some models using all measurements within the subset identified above, while others used only some of these measurements. Unless specified otherwise, all results were obtained using only the data from steady-state conditions.

The initial analysis focused on predicting both plant power output and other sensor data over different prediction time horizons. Different model configurations and training data were used with the NAR model, yielding the results provided in Table 1. The highlighted rows in the table indicate the case in which the data split was 80/20, as discussed earlier (Section 3.1). For comparison, results are also given (rows without highlighting) from when the training data consisted of all data from refueling cycles with similar plant output on average, with data from other refueling cycles used as test data to evaluate prediction performance.

The number of inputs indicates the number of principal components selected for the NAR model. Table 1 shows that, in general, too many nodes can yield just as poor generalization as too few nodes. Increasing the number of inputs does not always increase the performance of the model. By using fewer PCs, the model sees less noise that is uncorrelated from the signals of interest. The results also indicate a dramatic reduction in the prediction error when using an 80-20 data split (highlighted rows). An examination of the data indicated a small but measurable and consistent change in the generated output between the refueling cycles corresponding to the test and training data. The result of this difference (likely due to an increase in the plant generation capacity in this cycle) was a systematic bias in the predictions. This bias was alleviated by introducing an 80-20 split in the training data, so that the training data consisted of parts from all fueling cycles. This initial result also highlighted the need to ensure the training data is representative when using prognostic models that learn from the data.

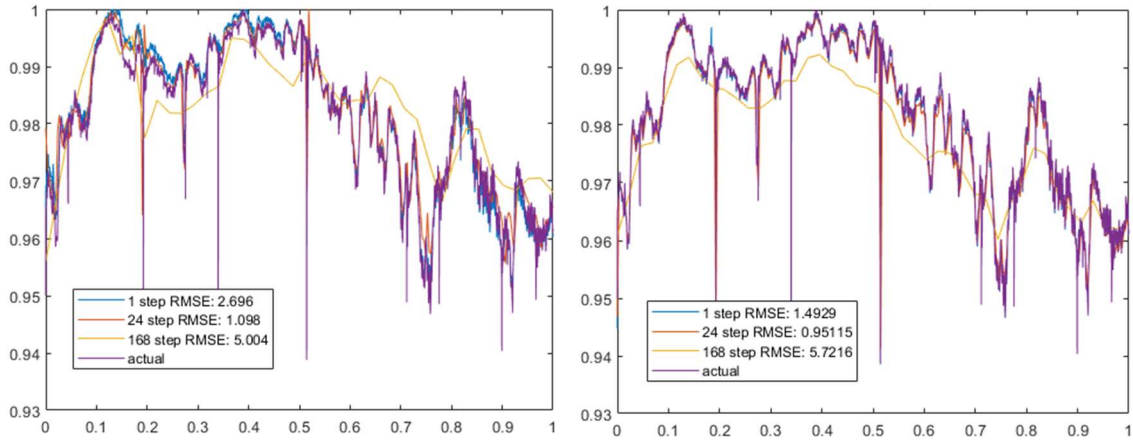
Table 1. Summary of the NAR model performance under different combinations of nodes, delays, inputs, and training data. RMSE was averaged for all signals estimated.

# of nodes	# of delays	# of inputs	RMSE
10	10	25	2.068
10	2	25	2.110
25	2	25	1.694
35	2	25	0.881
50	2	25	1.759
35	2	78	2.310
35	2	16	0.660
35	2	13	0.460
35	2	10	0.404
35	2	16	1.711
10	2	6	0.167
12	**2**	**6**	**0.087**
16	2	6	0.126

Figure 3(a) shows an example of the prediction results for the 12-node, 2-delay, 6-input NAR model using an 80/20 training/test data split, for three prediction horizons. The figure presents the normalized load (power generated) against time (normalized as a fraction of plant operation time). Separate NAR models were used to predict one hour, one day, and one week ahead. Figure 3(b) shows similar results for the SVR model. This SVR had a linear kernel function and used default settings resulting in 857, 692, and 146 support vectors, respectively. Parameter optimization will be necessary to optimize the number of support vectors and enhance the generalization ability of the SVR. For the LSTM models, the same 80/20 split was used. The results summarized in Figure 4 (varying prediction time horizons for LSTM models with different numbers of hidden nodes) appear to show a decrease in prediction accuracy with an increasing prediction time horizon. However, the data indicate variation in this trend, with a potential increase in accuracy as the horizon increases to about a day before decreasing again. A comparison against SVR models using linear and Gaussian kernels are also shown in Figure 4.

In the results presented in Figure 3 and Figure 4, the apparent increase in accuracy at the one-day prediction horizon as compared to the one-hour and one-week predictions was unexpected. This is likely a result of the data used in the training and testing and not a general finding on prediction accuracy as a function of prediction time horizon. Indeed, it would be expected that the prediction accuracy would be better when attempting to predict closer in time. The specific cause of the unexpected trend observed has not been determined, but likely causes include network size (for LSTM) and a fortuitous periodicity for small derates in which the 12 to 24-hour window helps the models avoid having to predict many derates using steady-state data, and vice versa. This latter potential cause is being investigated further to determine if this is an artifact of the sampling rates used to archive and provide the data.

It is worth noting that with all models, performance degradation was seen if the output was routed back to the input to enable multiple time step prediction (i.e., using the predicted output at a time step to predict the output at the next time step), as a result of prediction error accumulating over time. Such a scenario might occur when a single model is operating in real time to obtain multiple-time horizon predictions. In contrast, the use of a separate model specifically trained to predict one day or one week in advance appears to provide better results.



(a) (b)

Figure 3. Summary of predictions for the test set (steady-state operations) from (a) NAR model with 12 nodes, and (b) SVR model. Each plot shows the prediction performance for three different prediction time horizons. The horizontal axis is normalized as a fraction of plant operation time.

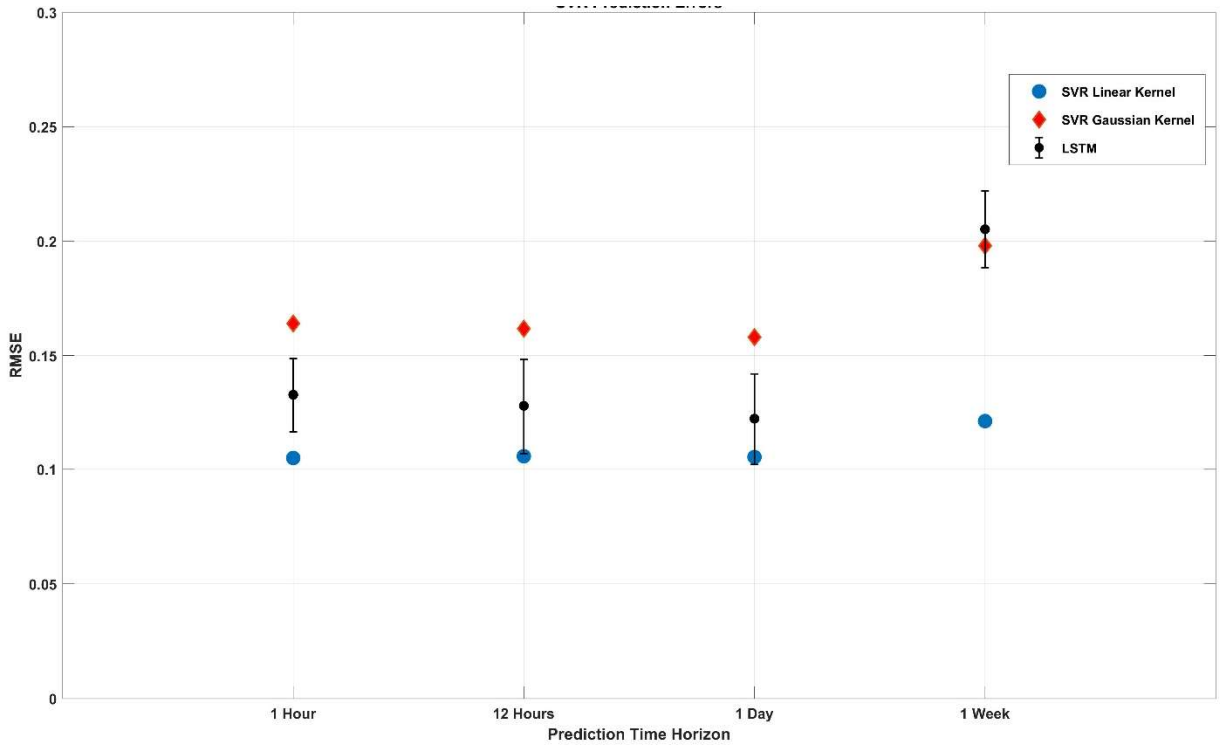
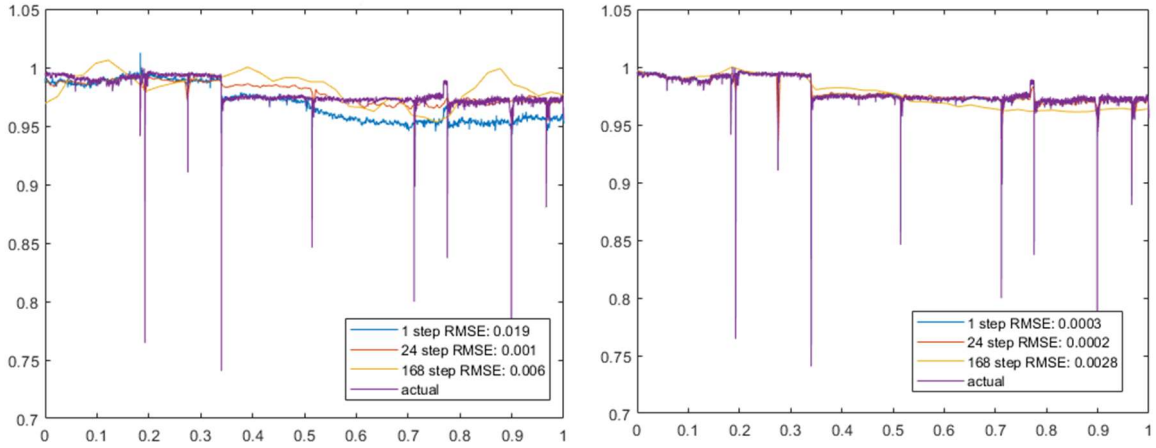


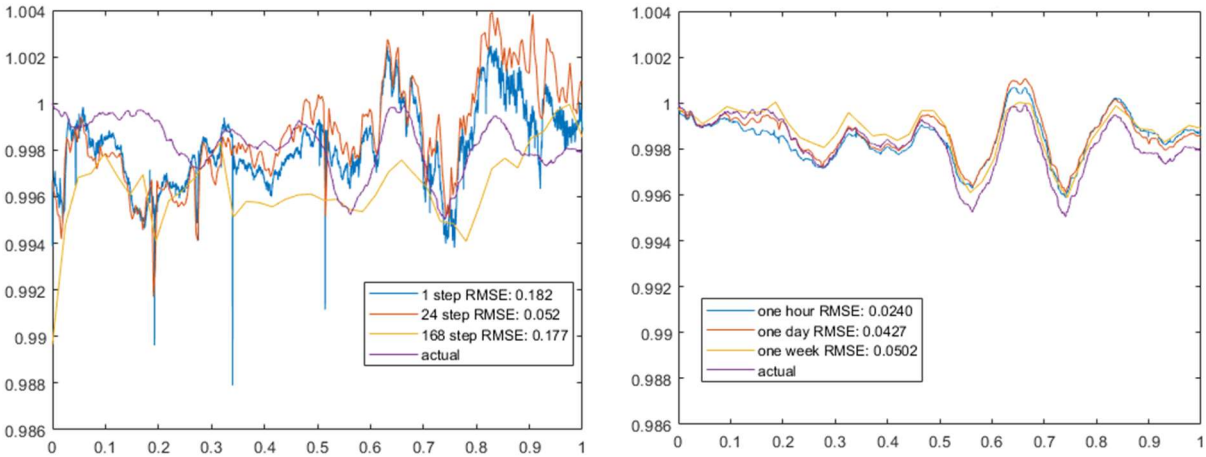
Figure 4. Comparison of RMSE of predictions for the test set (steady-state operations) from LSTM and SVR models as a function of the prediction horizon (number of time steps ahead of which the prediction is made). For each prediction horizon (1 hour, 12 hours, 1 day, and 1 week), the variation in RMSE for the LSTM models are due to models with different numbers of hidden layer nodes.

The models were also used to predict the flow in a single reactor loop for three different prediction horizons. In Figure 5, the NAR prediction is compared against an SVR model (5903, 411, and 89 support vectors for the three prediction horizons, respectively). Again, the one-day prediction appears to outperform both the one-hour and one-week predictions. However, the NAR model in Figure 5 shows that a consistent bias developed halfway through the test. The cause of this deviation is currently unknown but is being investigated. The SVR outperformed the NAR model by roughly an order of magnitude for each prediction horizon. In Figure 6, similar results were seen when comparing the prediction performance for the reactor feedwater pump temperature of NAR and SVR models (19, 20, and 47 support vectors, for the three prediction horizons for SVR). A similar performance improvement is seen when comparing the LSTM and SVR models (Figure 7); the performance improvement is particularly significant at longer prediction time horizons. However, the prediction performance of the non-optimized SVR also varies as it is applied to predict different variables (Figure 11).



(a) (b)

Figure 5. Summary of flow predictions for three different prediction horizons for the test set (steady-state operations) from (a) 12-node NAR model, and (b) SVR model, as a function of a million gallons per minute for a fraction of plant operation time.



(a) (b)

Figure 6. Summary of predictions for three different prediction horizons for the reactor feedwater pump temperature (steady-state operations) from (a) NAR model, and (b) SVR for a fraction of plant operation time.

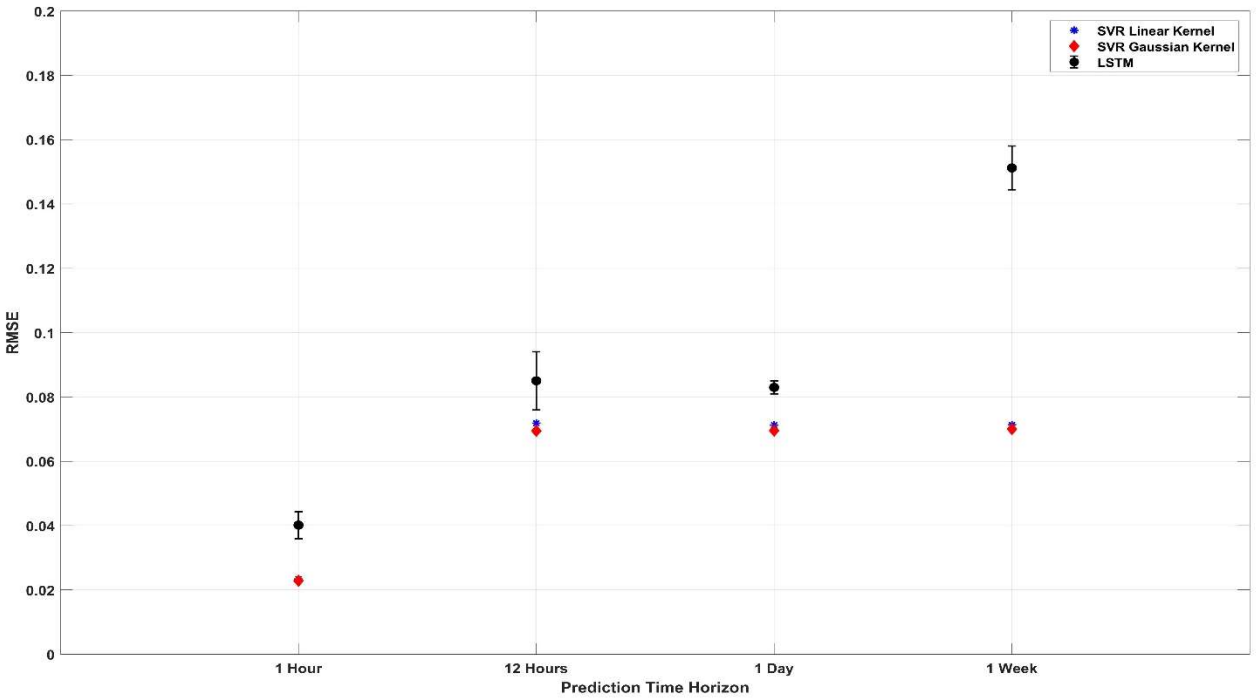


Figure 7. Summary of average flow predictions for the test set (steady-state operations) from LSTM and SVR models as a function of the prediction time horizons.

While the apparent performance advantage for the SVR could be due to the inherent advantages of the model itself, it is more likely that the performance gain is due to sub-optimal model structure for the NAR and LSTM models. The result appears to indicate a need for model structure optimization for LSTM and NAR models and likely for all machine-learned models, where model structure selection is a key part of model development.

Figure 8 and Figure 9 show examples of the performance variation of LSTM as a function of the number of hidden layer nodes. In the case of data shown in Figure 9, the LSTM was trained with a single input corresponding to data from one sensor within a moving time window to predict the value from the same sensor at a future time instant (1 hour, 12 hours, 24 hours, or 1 week in the future). Figure 8 illustrates the more complex situation corresponding to prediction of gross load at a future time instant from a set of process measurements. In both instances, the variation is largely due to variation in accuracy at different prediction time horizons. As can be seen from these examples, the prediction accuracy is a complex function of the hidden layer size (larger hidden layers are not always better) and the prediction time horizon.

Whereas the hidden layer size (number of LSTM cells) is related to the amount of history from which the LSTM network learns, often there is a limit beyond which increasing the size of the hidden layer brings diminishing returns. Unfortunately, there appears to be no simple way to identify this limit; nor is there a monotonic relationship between the number of hidden layer nodes and the performance of the network before this limit. There is also evidence in the literature that LSTMs do not scale well with data size, indicating that the use of LSTMs as prognostic models for predicting over a very long time series may be challenging.

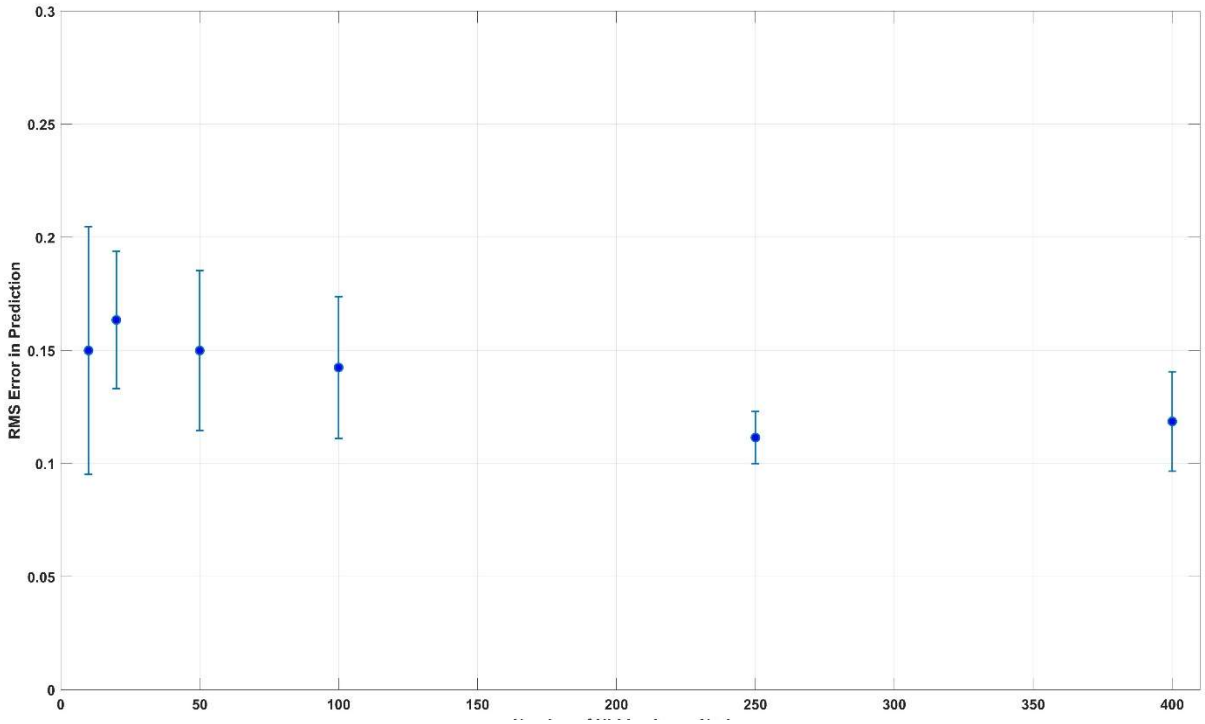


Figure 8. Performance variation of LSTM with hidden layer size. The LSTM was trained to predict the gross load under steady-state conditions using previous measurements of process variables (flow, temperature, pressure). The variation in RMSE for each hidden layer size is due to the different prediction time horizons: higher RMSE usually correlates with longer prediction horizon in this instance.

Similar model structure optimization can be performed for SVR by modifying the underlying kernel functions and the kernel bandwidth, as well as adjusting other algorithm parameters. An example of the performance obtained when using a different kernel function (Gaussian), along with optimizing the SVR model (number of support vectors, etc.), is shown in Figure 10. (For reference, the previous SVR results were obtained using linear kernels).

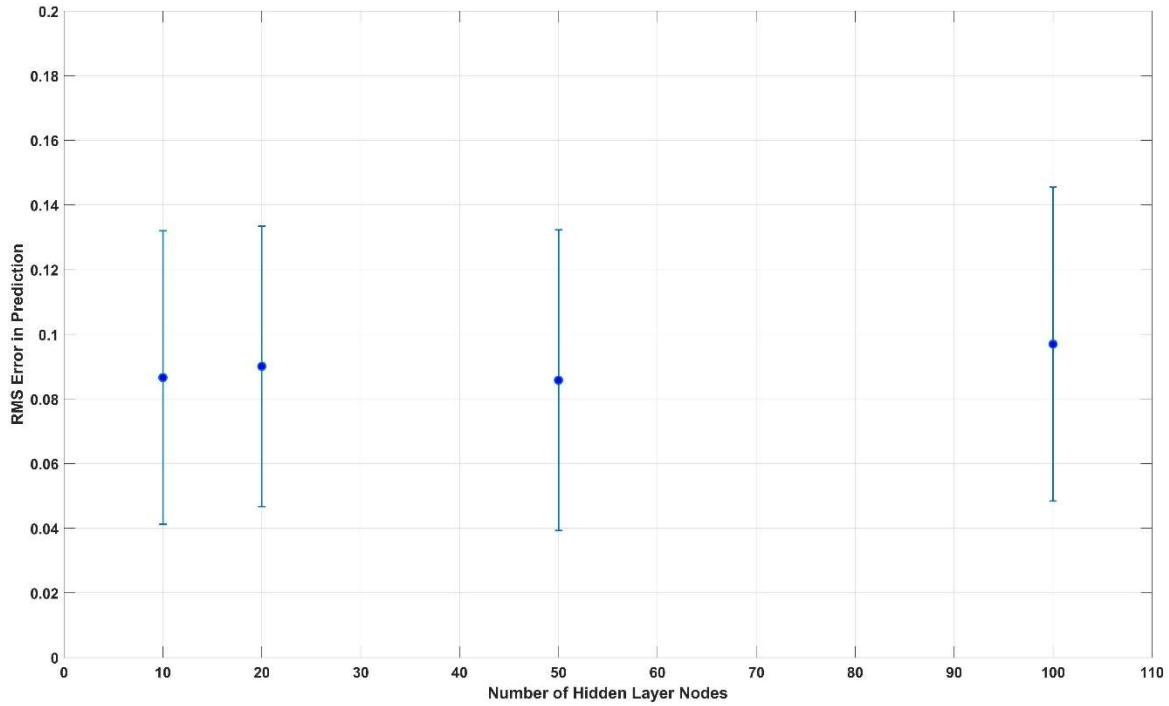


Figure 9. Performance variation of LSTM with hidden layer size. The LSTM was trained to predict the average flow under steady-state conditions using previous measurements of total flow. The variation in RMSE for each hidden layer size is due to the different prediction time horizons: higher RMSE usually correlates with longer prediction horizon in this instance.

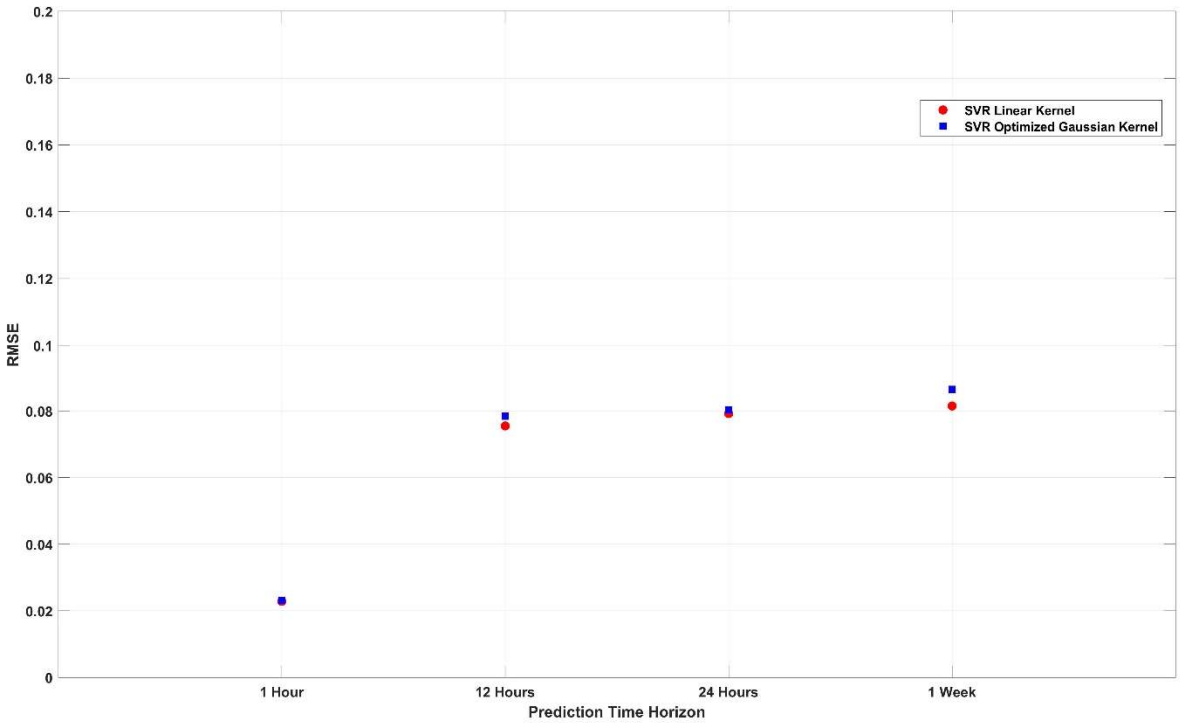


Figure 10. Prediction performance when using optimized Gaussian kernels in the SVR algorithm.

A related issue evident when analyzing Figure 3 through Figure 6 is the need to assess whether the information in the data allows for improved prediction results. This question was evaluated by applying the SVR model to predict the measurement from each sensor using only previous data from the same sensor. One SVR model per sensor was used, and the results are summarized in Figure 11 as the mean and standard deviation of the prediction RMSE across all sensors. The results are mixed, with the greatest variation in RMSE at intermediate time horizons indicating increased accuracy in some cases over the shorter prediction horizons. This reinforces the "fortuitous periodicity" theory in which not all data show the best prediction performance at intermediate prediction horizons, and it illustrates the need to quantify the prognosability of the available data as an important step prior to model building. The data in Figure 11 are plotted on the same scales that were used in Figure 7; it can be readily seen that the overall RMSE in all instances is better than that from the LSTM, but that the LSTM approaches the SVR results at intermediate prediction horizons. This result further indicates a potential need for model structure optimization for LSTM and SVR models.

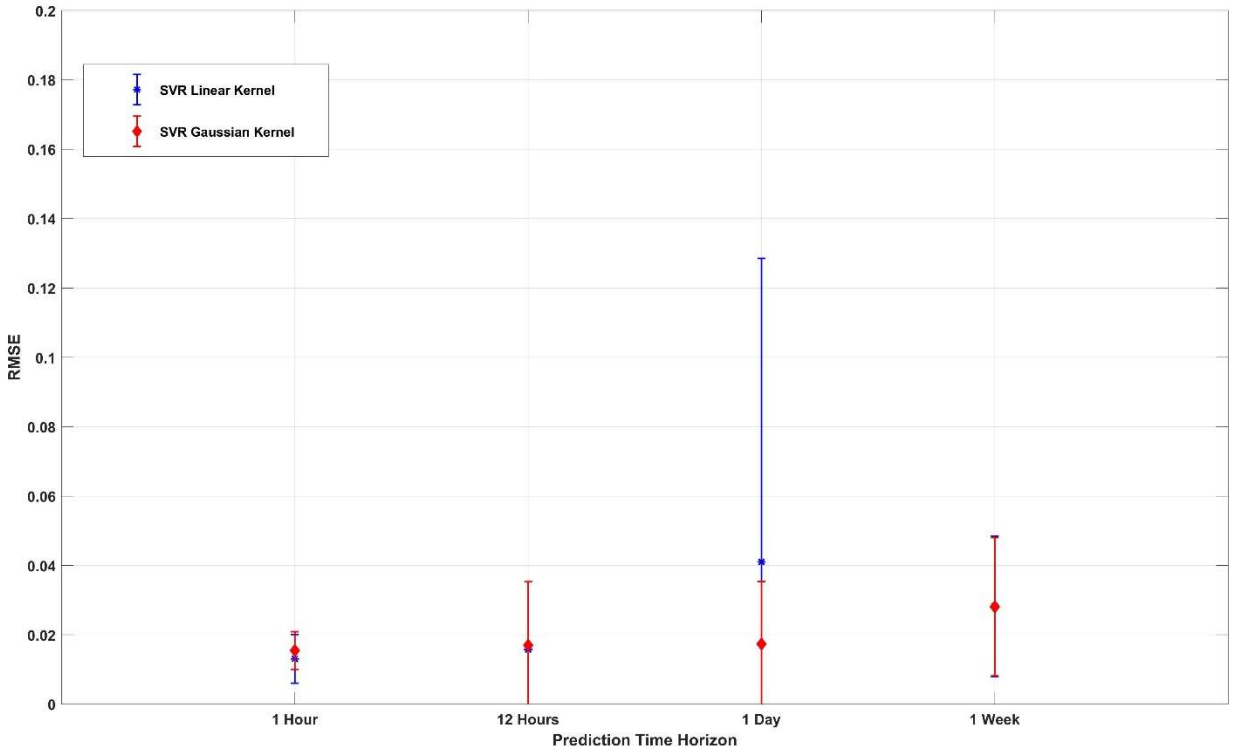


Figure 11. Summary of SVR prediction accuracy for single variable prediction, computed across multiple variables, for the test set (steady-state operations) as a function of the prediction time horizons.

Finally, Figure 12 illustrates the prediction performance when attempting to train an LSTM model to predict under both steady-state and transient conditions. In this case, the models were trained to predict the power output from the reactor one time step in advance using the feedwater flow values. Figure 12 shows a portion of the time window over which prediction was attempted, highlighting both the steady state and derates. While this is a hybrid of the two plant condition prognostics sub-problems presented earlier, nonetheless, the results are useful for drawing somewhat broad conclusions about the performance of the algorithm. Specifically, as evident from the performance, the model appears to be capable of learning what appears to be a relatively straightforward relationship between the input and output variables, even if separated in time. The relatively stable performance in steady state regions is probably a

result of the fact that the reactor power (or any process variable value) one time step ahead is not likely to be very different from its values at the present instant. The predictions are also seen to track derates and were observed to also track the ramp down prior to a refueling outage. This is also likely due to the fact that change in power and plant conditions is small one-time-step ahead. However, the prediction seems to have a bias, the cause of which is not entirely clear. While this may be the result of various trips/derates included in the data used for training, the specific impact of these transient conditions on the prediction performance should be better quantified. The performance under these conditions of other models also should be quantified to be able to draw broader conclusions about the ability to build accurate prognostic models that are capable of learning transient behavior.

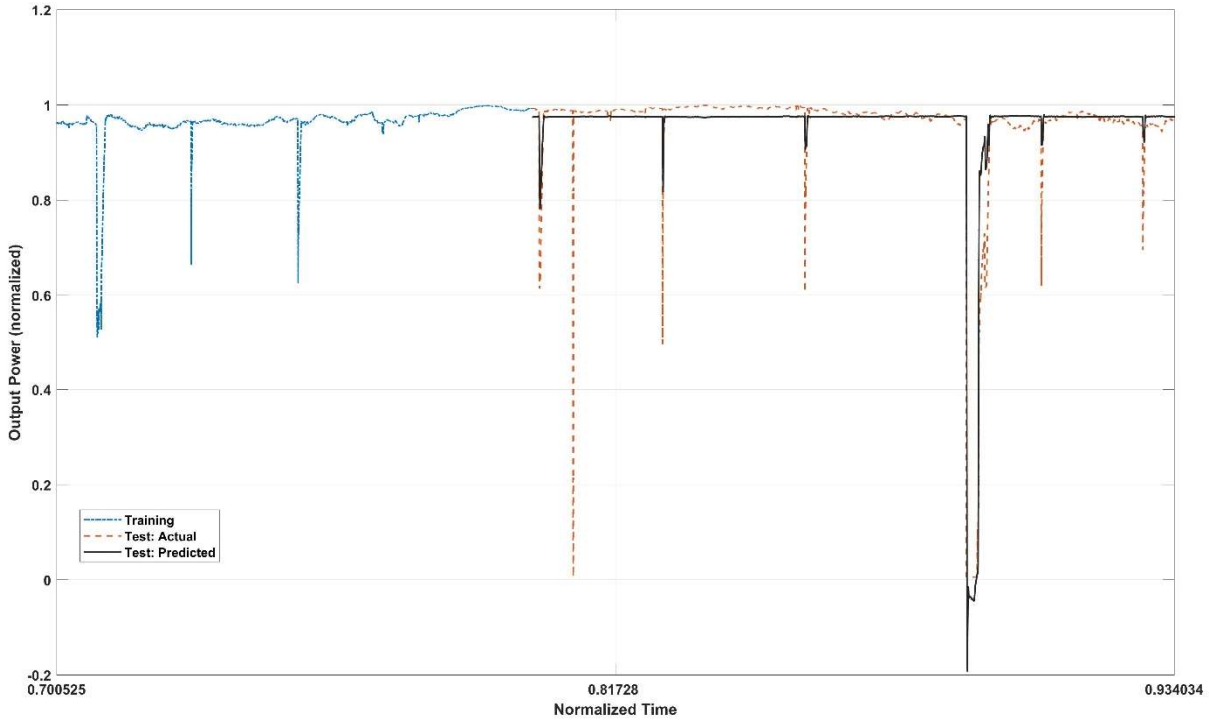


Figure 12. LSTM model performance for prediction of transient operations. The LSTM was trained to predict the gross load using previous measurements of process variables (flow, temperature, pressure) one time step into the future.

3.3.2 Discussion

While the assessment was not comprehensive, as it did not examine the effect of each parameter systematically, it did explore a large range within the parameter space, and the findings may be generalizable as a result. The results indicate that, as expected, the time required to train the models increases as the model complexity increases. However, the initialization of the LSTM and NAR parameters was also observed to influence the training time: initialization closer to an optimal solution tends to result in faster convergence. Also obvious from the training process was the fact that the actual training time depends on the data and the algorithm convergence parameters: loss function value at which to stop training, maximum number of epochs for training, mini-batch size, sequence length, etc.

Under steady-state conditions, the accuracy of short-term predictions (1 hour – 1 day) tends to be high, with an RMSE less than about 1 and generally much lower, indicating that the underlying system is relatively stable over the short term. The accuracy degrades as the prediction horizon increases, with the predictions for one week out tending to be worse than those for one day out in most cases. Model

structure optimization is considered crucial to prediction performance optimization; however, the performance is also considered to be dependent on the prognosability from the input data.

While the actual performance is a function of the model structure and the specific data used as input, the general performance degradation with increasing prediction horizon is not surprising, as other factors besides the data used in the prognostic model development influence the actual process variable values and the plant output. These factors include operational changes and degradation in other parts of the plant outside the FWCS that have an impact on plant generation and in turn affect the FWCS measurements, as well as the potential slow aging of the FWCS components.

A specific example is when reduction in load demands may be accompanied by a set of condensate and condensate booster pumps being taken offline. The FWCS is one of many subsystems in a nuclear power plant. In the specific plant from which data was obtained for this study, many of the derates, trips, and other performance issues were not a result of the feedwater condensate system but were due to operational changes in other systems, operator control actions, periodic surveillance activities that resulted in a subsystem configuration change, or from external factors. To accurately determine if the component was taken offline due to maintenance or performance concerns, this component outage would need to be confirmed with operator performance logs. These results appear to indicate that the use of prognostic models for plant condition prediction are appropriate for short time horizons; predicting over longer time horizons is likely to require additional data from multiple systems that may be difficult to obtain and integrate.

It is important to also differentiate between the performance obtained from a single model specifically trained to predict the output at a fixed time horizon (such as 1 day) and the performance obtained by using the output of a model trained to predict one hour in advance in an iterative fashion to achieve prediction by a day or more. In the latter case, prediction errors tend to accumulate as the model output is fed back to the input; the consequence is usually greater error in the predictions. In these instances, the prognostic models must be augmented with mechanisms that update the prediction as new measurements become available. This type of update has the consequence of calibrating the predictions and lowering the prediction error, at least temporarily. Potential mechanisms for updating predictions based on new measurements have been proposed in the literature [39] and include Kalman filters and particle filters [40]. In this report, such augmentation of the prognostic methods is being studied as part of the ongoing research. The results described earlier explicitly indicate which of the two cases was used in estimating the prediction accuracy.

The model performance was found to be sensitive to normalization of the inputs, as well as to whether features—PCA in this instance—were used or not. However, the performance gains with PCA as a preprocessing tool were inconsistent across model types. It is also worth noting that there are several combinations possible for the inputs and outputs to the models, and to a great extent, the proper choice of which sensors to use as inputs (predictors) tended to be a matter of trial and error given this data set. Alternative metrics, such as the Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC), are necessary to quantify the information gain from a specific input (predictor) and objectively determine the subset of predictors that should be selected for prognostics.

A key finding here for degradation prognostics is the need for data from the system under monitoring, as well as other associated data from the plant process that may impact performance of the system being monitored. A related issue is the selection of appropriate prognostic models. While machine learning models appear to be quite capable of prognosing the quantity of interest, relevant labeled data must to be available for training the models. Furthermore, the model structure (number of nodes, neural network connectivity, etc.) does not appear to have a monotonic relationship with the prediction accuracy, although there appears to be a monotonically increasing relationship between network size and

computational complexity, as in the time to train the network to reach an acceptable training error and the associated memory requirements. As such, the use of these models is expected to require some trial and error before an acceptable model structure and performance can be achieved.

The expected increases in prediction errors as the prediction horizon is increased point toward the need to quantify the confidence associated with the prognostic result. This is a generalizable result to degradation prognostics and is an ongoing research effort, with further results to be documented in subsequent reports.

4. SUMMARY AND PATH FORWARD

Prognostic models for nuclear power plants are expected to enable data-driven insights into plant performance that can drive condition-based predictive maintenance scheduling. If properly applied, prognostic models are a key enabling technology, optimizing plant performance using the right type of data and information. Advances in machine learning and related data analysis methodologies have opened up opportunities for using operational data to derive prognostic models for assessing system state and detecting anomalies in equipment condition. A series of studies using three types of exemplar models and data from an operating plant indicates that the models can learn relatively quickly and may be used to predict the process measurement of interest over different time horizons. The prediction accuracy is dependent on the model structure and other factors, including the data used in the training process, with the accuracy degrading as the time horizon increases. Results indicate that the prediction accuracy tends to be highest when working with data from steady-state conditions; the use of data that includes transients challenges the ability of the model to predict the behavior during and sometimes after the transient as occurred. It is anticipated that this is due to the lack of sufficient data capturing the conditions before, during, and after the transient, as well as the fact that the available data from the FWCS does not appear to contribute to the various transient behaviors seen (derates, trips, ramp-down and start-up).

The analysis to date indicates a need to curate and clean data sets for use in similar applications; a need for data covering equipment degradation or faults; and a need to quantify the prediction uncertainty. The ability to update prognostic models as relevant data become available is also a necessity. These questions are the subject of ongoing research in this project.

5. ACKNOWLEDGEMENTS

This report was made possible through funding by the U.S. Department of Energy's Office of Nuclear Energy under the Nuclear Energy Enabling Technologies Program. We are grateful to Suibel Schuppner at the U.S. Department of Energy and Craig Primer at Idaho National Laboratory for championing this effort. We also thank Rose Raney for her assistance in technical editing and formatting of the document and the technical reviewers for their feedback. The data used in this research was provided by Exelon Generation Company. We are also thankful to the plant engineers for their technical discussions on the data.

REFERENCES

[1] V. Agarwal, "Wireless sensor modalities at a nuclear power site to collect vibration data," Idaho National Laboratory, Idaho Falls, ID, May 2020.

[2] K. A. Manjunatha and V. Agarwal, "Techno-economic analysis framework for wireless networks in nuclear power plants," Idaho National Laboratory, Idaho Falls, ID, September 2019.

[3] V. Agarwal, "Enabling predictive maintenance with wireless instrumentation in balance of plant system," Idaho National Laboratory, Idaho Falls, July 2019.

[4] K. A. Manjunatha and V. Agarwal, "Review of wireless communication technologies and techno-economic analysis," Idaho National Laboratory, Idaho Falls, ID, May 2019.
[5] R. M. Meyer *et al.*, "Technical Needs for Prototypic Prognostic Technique Demonstration for Advanced Small Modular Reactor Passive Components," Pacific Northwest National Laboratory, Richland, WA, PNNL-22488 Rev. 0, SMR/ICHMI/PNNL/TR-2013/01, 2013.
[6] J. Coble, P. Ramuhalli, L. J. Bond, J. W. Hines, and B. Upadhyaya, "A Review of Prognostics and Health Management Applications in Nuclear Power Plants," *Int'l. J. Progn. Health Manage.,* vol. 6, 16, p. 22, 2015.
[7] M. Kantor, L. Wajda, B. Lannoo, L. Casier, and e. al., "General Framework for Techno-Economical Analysis of Next Generation Access Networks," presented at the 12th International Conference on Transparent Optical Networks, Munich, Germany, June/July 2010, 2010.
[8] V. Agarwal, N. Lybeck, B. Pham, R. Rusaw, and R. Bickford, "Asset Fault Signatures for Prognostic and Health Management in the Nuclear Industry," *IEEE Reliability Digest,* February 2015 2015.
[9] V. Agarwal, N. Lybeck, B. Pham, R. Rusaw, and R. Bickford, "Prognostic and Health Management of Active Assets in Nuclear Power Plants: Special Issue on Nuclear Energy PHM," *International Journal of Prognostics and Health Management,* vol. 6, pp. 1-17, 2015.
[10] P. Ramuhalli, J. W. Griffin, J. M. Fricke, and L. J. Bond, "An Assessment of Uncertainty in Remaining Life Estimation for Nuclear Structural Materials," in *8th International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies 2012, NPIC and HMIT 2012: Enabling the Future of Nuclear Energy, July 22-25, 2012, San Diego, California*, LaGrange Park, IL, 2012-12-01/ 2012, vol. 2: American Nuclear Society, pp. 1325-1337.
[11] M. A. Khaleel and F. A. Simonen, "Evaluations of Structural Failure Probabilities and Candidate Inservice Inspection Programs," U.S. Nuclear Regulatory Commission, Washington, D.C., 2009.
[12] F. A. Simonen *et al.*, "Probabilistic Fracture Mechanics Evaluation of Selected Passive Components – Technical Letter Report," Pacific Northwest National Laboratory, Richland, Washington, 2007.
[13] Y. A. Yucesan and F. A. C. Viana, "A physics-informed neural network for wind turbine main bearing fatigue," *International Journal of Prognostics and Health Management,* vol. 11, 1, 2020.
[14] P. C. Paris and F. Erdogan, "A critical analysis of crack propagation laws," *Journal of Basic Engineering,* 1963.
[15] M. Baur, P. Albertelli, and M. Monno, "A review of prognostics and health management of machine tools," *Int. J. Adv. Manuf. Technol.,* vol. 107, no. 5–6, pp. 2843–2863, 2020, doi: 10.1007/s00170-020-05202-3.
[16] D. An, N. H. Kim, and J. H. Choi, "Practical options for selecting data-driven or physics-based prognostics algorithms with reviews," *Reliab. Eng. Syst. Saf.,* vol. 133, pp. 223–236, 2015, doi: 10.1016/j.ress.2014.09.014.
[17] M. Kayri, "Predictive abilities of Bayesian regularization and levenberg-marquardt algorithms in artificial neural networks: A comparative empirical study on social data," *Math. Comput. Appl.,* vol. 21, no. 2, 2016, doi: 10.3390/mca21020020.
[18] A. Diez-Olivan, J. Del Ser, D. Galar, and B. Sierra, "Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0," *Inf. Fusion,* vol. 50, pp. 92-111, 2019, doi: 10.1016/j.inffus.2018.10.005.
[19] A. Dourado and F. A. C. Viana, "Physics-informed neural networks for missing physics estimation in cumulative damage models: a case study in corrosion fatigue," *ASME Journal of Computing and Information Science in Engineering,* vol. 20, no. 6, p. 10, 2020, doi: 10.1115/1.4047173.
[20] O. F. Eker, F. Camci, and I. K. Jennions, "A new hybrid prognostic methodology," *Int. J. Progn. Heal. Manag.,* vol. 10, pp. 1-13, 2019.

[21] M. S. Kan, A. C. C. Tan, and J. Mathew, "A review on prognostic techniques for non-stationary and non-linear rotating systems," *Mechanical Systems and Signal Processing,* vol. 62-63, pp. 1-20, 2015/10/01/ 2015, doi: https://doi.org/10.1016/j.ymssp.2015.02.016.
[22] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mechanical Systems and Signal Processing,* vol. 107, pp. 241-265, 2018/07/01/ 2018, doi: https://doi.org/10.1016/j.ymssp.2017.11.024.
[23] M. Schwabacher and K. Goebel, "A Survey of Artificial Intelligence for Prognostics," in *Artificial Intelligence for Prognostics: Papers from the AAAI Fall Symposium*, Arlington, Virginia, November 9-11, 2007 2007: AAAI Press, Menlo Park, California, pp. 107-114.
[24] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation,* vol. 9, no. 8, pp. 1735-1780, 1997, doi: 10.1162/neco.1997.9.8.1735.
[25] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network," *IEEE Trans. Smart Grid,* vol. 10, no. 1, pp. 841-851, 2019, doi: 10.1109/TSG.2017.2753802.
[26] A. Farzad, H. Mashayekhi, and H. Hassanpour, "A comparative performance analysis of different activation functions in LSTM networks for classification," *Neural Computing and Applications,* vol. 31, no. 7, pp. 2507-2521, 2019/07/01 2019, doi: 10.1007/s00521-017-3210-6.
[27] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Trans. Neural Networks Learn. Syst.,* vol. 28, no. 10, pp. 2222-2232, 2017, doi: 10.1109/TNNLS.2016.2582924.
[28] K. Benmouiza and A. Cheknane, "Forecasting hourly global solar radiation using hybrid k-means and nonlinear autoregressive neural network models," *Energy Convers. Manag.,* vol. 75, pp. 561-569, 2013, doi: 10.1016/j.enconman.2013.07.003.
[29] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, " Support vector regression machines," *Advances in neural information processing systems,* pp. 155-161, 1997.
[30] B. Dong, C. Cao, and S. E. Lee, "Applying support vector machines to predict building energy consumption in tropical region," *Energy Build.,* vol. 37, no. 5, pp. 545-553, 2005, doi: 10.1016/j.enbuild.2004.09.009.
[31] R. K. Jain, K. M. Smith, P. J. Culligan, and J. E. Taylor, "Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy," *Appl. Energy,* vol. 123, pp. 168-178, 2014, doi: 10.1016/j.apenergy.2014.02.057.
[32] V. Climente-Alarcon, J. A. Antonino-Daviu, E. G. Strangas, and M. Riera-Guasp, "Rotor-bar breakage mechanism and prognosis in an induction motor," *IEEE Trans. Ind. Electron.,* vol. 62, no. 3, pp. 1814–1825, 2015, doi: 10.1109/TIE.2014.2336604.
[33] R. Li, W. J. C. Verhagen, and R. Curran, "Comparison of Data-driven Prognostics Models: A Process Perspective," presented at the 29th European Safety and Reliability Conference, 2019.
[34] A. Oluwasegun and J. C. Jung, "The application of machine learning for the Prognostics and Health Management of control element drive system," *Nucl. Eng. Technol.,* vol. 52, no. 10, pp. 2262–2273, 2020, doi: 10.1016/j.net.2020.03.028.
[35] J. Coble and J. W. Hines, "Identifying optimal prognostic parameters from data: a genetic algorithms approach," presented at the Annual Conference of the Prognostics and Health Management Society PHM, San Diego, CA, 2009.
[36] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction,," *Mech. Syst. Signal Process..,* vol. 104, pp. 799–834, 2018, doi: 10.1016/j.ymssp.2017.11.016.
[37] N. S. Rao, C. R. Greulich, P. Ramuhalli, M. S. Cetiner, and P. Devineni, "Sensor Drift Estimation for Reactor Systems by Fusing Multiple Sensor Measurements," presented at the 2019 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC) Manchester, UK, 2020.

[38] N. S. Rao, P. Ramuhalli, C. R. Greulich, and M. S. Cetiner, "Sensor Error Estimation for Reactor Coolant System with Generalization Error Equations," presented at the 2019 ANS Winter Meeting and Nuclear Technology Expo 2019.

[39] P. Ramuhalli, L. J. Bond, J. W. Griffin, M. Dixit, and C. H. Henager, Jr., "A Bayesian Prognostic Algorithm for Assessing Remaining Useful Life of Nuclear Power Components," in *7th International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies (NPIC&HMIT 2010)*, Las Vegas, Nevada, November 7-11, 2010 2010: American Nuclear Society, LaGrange Park, Illinois, pp. 875-886, Vol. 2.

[40] M. E. Orchard and G. J. Vachtsevanos, "A Particle Filtering-based Framework for Real-time Fault Diagnosis and Failure Prognosis in a Turbine Engine," in *2007 Mediterranean Conference on Control and Automation*, Athens, Greece, July 27-29, 2007 2007: IEEE Computer Society, Piscataway, New Jersey in 2007 Mediterranean Conference on Control and Automation, MED, p. 4433871, doi: 10.1109/med.2007.4433871. [Online]. Available: http://dx.doi.org/10.1109/MED.2007.4433871