Abstract—With the rapid development of sensor and information technology, now multisensor data relating to the system degradation process are readily available for condition monitoring and remaining useful life (RUL) prediction. The traditional data fusion and RUL prediction methods are either not flexible enough to capture the highly nonlinear relationship between the health condition and the multisensor data or have not fully utilized the past observations to capture the degradation trajectory. In this article, we propose a joint prognostic model (JPM), where Bayesian linear models are developed for multisensor data, and an artificial neural network is proposed to model the nonlinear relationship between the residual life, the model parameters of each sensor data, and the observation epoch. A Bayesian updating scheme is developed to calculate the posterior distributions of the model parameters of each sensor data, which are further used to estimate the posterior predictive distributions of the residual life. The effectiveness and advantages of the proposed JPM are demonstrated using the commercial modular aero-propulsion system simulation data set.

Index Terms—Bayesian inference, degradation modeling, joint prognostic model (JPM), neural network (NN), remaining useful life (RUL) prediction.

I. INTRODUCTION

MAINTAINING high reliability in modern engineering systems, e.g., airlines and automobiles, is essential to achieve a desirable efficiency and productivity. As a result, there is a growing need for prognostics to eliminate unscheduled breakdowns and reduce maintenance costs. Prognostics refer to the estimation of the remaining useful life (RUL) of degrading systems and components based on the current health condition [1]. In general, there are two types of prognostic models: physical-based and data-driven models [2]. For physical models, it is necessary to fully understand the specific degradation mechanisms, which may not be feasible in practice due to limited knowledge or high system complexity [3]. In contrast, the data-driven models make use of the condition monitoring (CM) data that are closely related to degradation processes for RUL prediction. With the rapid development of information and sensing technology, now the data-driven models have been widely used in modern engineering systems for CM and predictive maintenance. Typical CM signals used in data-driven prognostics include temperature, vibration, and fuel consumption data collected during the system operation [1].

There are a large number of data-driven models in the existing literature. They can be grouped into two categories: artificial intelligence (AI) techniques and statistical approaches [2], [4]–[9]. AI techniques include the artificial neural networks (ANNs), support vector machines (SVMs), fuzzy logic systems, fuzzy-NNs, and evolutionary algorithms. Statistical approaches include various stochastic processes, state-space models, and regression-based models. Most of the existing models only focus on a single measure for RUL prediction. These approaches are effective when most of the underlying degradation characteristics can be well captured by a single sensor. In practice, however, a single signal may not be adequate to characterize the complex degradation process, especially in complicated systems. This may result in significant overestimation or underestimation of the remaining lifetime. In such cases, multiple sensors are needed to capture various aspects of the degradation process for prognostic improvement. Each sensor may contain only partial information about the same system. Some are highly related to the degradation mechanism, while others may not. Therefore, effectively fusing these sensor data is highly desirable and very promising to provide more accurate and robust prognostic results.

Data fusion-based prognostics have been intensively studied recently. They can be generally classified into two categories: 1) construction of a health index (HI) for condition assessment and RUL prediction and 2) establishing a functional mapping between the multisensor data and RUL. The methods of the first category are often unsupervised or semi-supervised. Liu et al. [1] developed a single composite HI by linearly combining different sensors to better characterize the degradation process. The linear combination is performed in such a way that it satisfies two essential properties, i.e., maximizing the monotonic property of the HI and minimizing the variance of the failure threshold. Later Liu and Huang [10] developed another HI via linear fusion of multiple degradation-based
signals and trained the model by minimizing both the model fitting errors and the variance of the failure threshold. However, these methods combine multisensor data linearly. In practice, due to system complexity, the sensor signals may have a highly nonlinear relationship with the degradation process, which will inevitably limit the effectiveness of the linear fusion methods. To overcome this issue, Song et al. [11] integrated the data fusion model with the kernel method to provide a nonlinear relationship between the HI and sensor signals. Nevertheless, the nonlinear relationship is limited greatly by the type of kernel functions. Therefore, more flexible nonlinear modeling is needed for prognostic improvement.

The prognostics of the second category are mainly supervised methods, where multisensor variables are used as or transformed into multidimensional features, and the corresponding RULs are used as labels. In this category, the ANN models are mostly used due to their adaptability, nonlinearity, and arbitrary function approximation ability [12]–[14]. Various ANN models with different network structures, input and output parameters, and activation functions have been developed for RUL prediction. Gebraeel et al. [15] proposed a feedforward NN-based approach for the RUL prediction of ball bearings. In their model, the vibration amplitudes of the first seven defective frequencies at the time of prediction were used as input and the residual life was used as the model output. However, due to unit-to-unit heterogeneity, using only the observation at the current time may not be effective for RUL prediction.

As shown in Fig. 1(a), unit 1 and unit 2 have the same degradation value at the time $t_1$, yet their degradation paths differ significantly, indicating a totally different residual life. If the observation at $t_1$ is used as input, the predicted residual life would be the same for both units, thus resulting in large error. Later, several improved ANN models were developed, where the raw observations or smoothed ones at both the previous and the current observation epochs, and the current epoch $t$ were used as model inputs, and the percentage of life occurred, or alternatively the remaining life percentage was used as model output [12], [16]–[18]. Using this strategy could better capture the health condition and degradation path. Nevertheless, the unit heterogeneity issue may still affect the prediction accuracy, which can be seen from Fig. 1(b), where two units with the same degradation values at $t_1$ and $t_2$ have significantly different degradation paths.

To more effectively capture the observed degradation path for RUL prediction, recurrent NN (RNN) and deep convolutional NN (CNN)-based methods are developed recently. Heimes [19] utilized an advanced RNN architecture to capture the long-term dependencies for RUL prediction. Wu et al. [20] proposed to use vanilla long short-term memory (LSTM) networks to make the most of LSTM ability for prognostic improvement. Babu et al. [21] developed a deep CNN-based approach for RUL prediction. In this method, the convolution and pooling filters are applied along the temporal dimension over the multisensor data. A similar method was also proposed in [22]. By utilizing a moving window of multiple successive observations as input, these methods can effectively mitigate the problem shown in Fig. 1. However, their performance is still not satisfactory in the early stage of the life cycles, where degradation is often not noticeable. The reason is that there is almost no change in the input features at the early stage, while the output RUL decreases linearly. In fact, all of these methods mentioned earlier used artificially labeled RUL instead of true RUL as the output. Within the early degradation stage, the RUL at a certain time epoch, e.g., the starting epoch of rapid degradation stage, was assigned to all previous time epochs, which is equivalent to starting to predict the RUL only when the obvious degradation appears. Therefore, this trick is actually not capable of predicting the RUL at the early stage.

To address the above-mentioned issues, we propose a joint prognostic model (JPM) in this article, where a Bayesian linear model is used to model the degradation signals, and an ANN is proposed to establish the relationship between the residual life and the model parameters of each sensor data. Specifically, a commonly used two-stage degradation modeling and CM scheme is proposed for each sensor signal [7], [23], [24]. At the offline stage, a Bayesian linear model is developed for each sensor based on historical data to capture both the population behavior and individual heterogeneity. At the online monitoring stage, a recursive Bayesian updating approach is applied to infer the model parameters of each sensor data sequentially for an in-service unit. To model the nonlinear relationship between the residual life and multisensor degradation data, a three-layer NN is developed, where the model parameters of multisensor data along with the time step $t$ are used as the input layer, and the occurred life percentage is used as model output. Since the online updated model parameters of the in-service unit are obtained by utilizing all the historical data of other units and the available data of that unit, they can more effectively capture the degradation path or the long-term dependence.

The remainder of this article is organized as follows. Section II introduces the technical details of the JPM framework, including the offline degradation modeling and estimation, the NN model formulation and training, and online Bayesian updating. Section III demonstrates and evaluates the effectiveness of the proposed JPM through a case study using National Aeronautics and Space Administration (NASA), Washington, DC, USA, commercial modular aero-propulsion system simulation (C-MAPSS) data set for commercial aircraft gas turbine engines [25]. The conclusion and discussion are provided in Section IV.

II. JOINT PROGNOSTIC MODEL FOR RUL PREDICTION

The overall framework of the JPM is shown in Fig. 2. There are two stages, offline stage and online stage. At the
Offline Stage

Historical data and model fitting for each signal
\( \beta_{ij} \)

MLE based prior estimation
\( \beta_{ij} \sim N(\mu_{ij}^{(0)}, \sigma_{ij}^{2,\text{MLE}}) \)
\( \sigma_{ij}^{2} \sim IG(\alpha_{ij}^{(0)}, \alpha_{ij}^{(0)}) \)

NN model formulation and training

Online Stage

Bayesian model updating

RUL prediction

Fig. 2. Illustration of the joint prognostic framework.

Offline stage, both the mixed-effects model for degradation modeling and the NN model for data fusion and RUL prediction are established. At this stage, a commonly used empirical two-step approach [8] is applied to estimate the distribution parameters of the mixed-effects model. In this approach, the model parameters of each degradation signal are obtained through the maximum likelihood estimation (MLE) for each unit, and then all the fit parameters are used to estimate the distribution parameters through the MLE method. In the NN model formulation and training, those estimated parameters of each degradation signal based on all the observations of the whole lifetime and observation epoch \( t \) are used as model input. At the online stage, a Bayesian model updating scheme is used to calculate the posterior distributions of degradation model parameters of an in-service unit, and then the posterior means are used as input of the NN model to estimate RUL. The technical details are provided in Sections II-A–II-C.

A. Bayesian Linear Modeling and Parameter Estimation

Mixed-effects models are often used to capture both the population behavior and unit-to-unit variability in degradation modeling. It can be generally formulated as [26]

\[ s_i = \eta(\alpha, \phi, t) + \epsilon_i \]  

where \( s_i \) is the measured degradation signal at time step \( t \), \( \eta \) is a parametric form of the degradation model, \( \alpha \) is a vector of the fixed-effects parameters, \( \phi \) is a vector of the random-effects parameters, and \( \epsilon_i \) is noise or error term following i.i.d. normal distributions. Typically, a linear model is assumed in the parametric form for its simplicity and flexibility. To account for the noise variance heterogeneity across different units, we propose to use a more general model, the Bayesian linear model, where the noise variance is also assumed random. Suppose there are \( J \) historical units, and each unit has \( J \) degradation signals. The following Bayesian linear model is proposed to characterize the multiple degradation signals

\[ s_{i,j,k} = X_{i,j,k} \beta_{i,j} + \epsilon_{i,j,k} \]  

where \( s_{i,j,k} \) is the sensor measurement for unit \( i \) and sensor \( j \) at inspection time \( t_{i,k} \), \( X_{i,j,k} \) is a \((q_j + 1)\)-dimensional vector of polynomial basis functions, i.e., \( X_{i,j,k} = [1, t_{i,k}, t_{i,k}^2, \ldots, t_{i,k}^{q_j}] \), \( \beta_{i,j} \) is a \((q_j + 1)\)-dimensional vector of regression parameters following a multivariate normal distribution, \( \epsilon_{i,j,k} \) is the i.i.d. measurement noise with \( \epsilon_{i,j,k} \sim N(0, \sigma_{i,j,k}^2) \), and \( \sigma_{i,j,k}^2 \) follows an inverse Gamma distribution. Note that here \( \sigma_{i,j,k}^2 \) is also assumed random to make it more flexible to capture the unit heterogeneity, which is different from the common way of using a deterministic noise level [1], [10]. To facilitate Bayesian updating at the online stage, a joint conjugate prior is assigned for \( \beta_{i,j} \) and \( \sigma_{i,j}^2 \), i.e., \( \beta_{i,j} \sim N(\mu_{i,j}^{(0)}, \sigma_{i,j}^{2,\text{MLE}}) \) and \( \sigma_{i,j}^2 \sim IG(\alpha_{i,j}^{(0)}, \alpha_{i,j}^{(0)}) \). Note that here we assume that all the multiple degradation signals of each unit are measured at the same time steps for notational convenience, i.e., \( t_{i,k}, k = 1, \ldots, n_i \) where \( n_i \) is the total number of inspection time steps for unit \( i \).

The model parameters of interest are \( \psi^{(j)} = \{\mu_{0,j}^{(j)}, \Sigma_{0,j}^{(j)}, \alpha_{1,j}^{(j)}, \alpha_{2,j}^{(j)}\}, j = 1, \ldots, J \), which are hyperparameters that will be used at the online stage for Bayesian updating. The MLE approach is a natural way to estimate \( \psi^{(j)} \). Specifically, \( \psi^{(j)} \) can be obtained by maximizing the following marginal likelihood:

\[ \hat{\psi}^{(j)} = \arg\max_{\psi^{(j)}} \prod_{i=1}^{J} \int \int p(s_{i,j} | \beta_{i,j}, \sigma_{i,j}^2) \pi(\beta_{i,j}, \sigma_{i,j}^2) d\sigma_{i,j}^2 d\beta_{i,j} \]  

where \( s_{i,j} = [s_{i,j,1}, s_{i,j,2}, \ldots, s_{i,j,m}]^T \) and \( \pi(\beta_{i,j}, \sigma_{i,j}^2) \) is the joint probability density function of \( \beta_{i,j} \) and \( \sigma_{i,j}^2 \). However, the above-mentioned marginal likelihood is very complex and not easy to optimize directly. A commonly used method to
handle this issue is the expectation-maximization (EM) algorithm [27], where $\beta_{i,j}$ and $\sigma_{i,j}^2$ can be treated as missing variables. However, the $Q$ function is intractable in the expectation step, and thus, the Monte Carlo-based EM algorithm may be required, which makes the problem very complex. To address this issue, we propose to use a two-step estimation approach, a much easier and efficient method where the degradation signals of each unit are fitted through the MLE method, and then the estimated parameters $\{\beta_{i,j}, \sigma_{i,j}^2\}$ are used to estimate hyperparameters $\psi^{(j)}$. Although some bias may be introduced, this bias is often negligible in the Bayesian online updating process [8].

Suppose $X_{i,j}$ is the design matrix for unit $i$ and sensor $j$, which is given as

$$\begin{bmatrix} 1 & t_{i,1} & \cdots & t_{i,1}^q & t_{i,2} & \cdots & t_{i,2}^q & \cdots & \cdots & \cdots & t_{i,n} & \cdots & t_{i,n}^q \end{bmatrix}$$ \hspace{1cm} (4)$$

then the MLE of $(\beta_{i,j}, \sigma_{i,j}^2)$ can be easily obtained as

$$\hat{\beta}_{i,j} = (X_{i,j}^T X_{i,j})^{-1} X_{i,j}^T s_{i,j}, \sigma_{i,j}^2 = \frac{\|s_{i,j} - X_{i,j} \hat{\beta}_{i,j}\|^2}{n_i}. \hspace{1cm} (5)$$

The hyperparameters $(\alpha_{i,j}^{(1)}, \alpha_{i,j}^{(2)})$ for the inverse-Gamma distributed noise variance can be estimated numerically by maximizing the likelihood function using estimated values $(\hat{\sigma}_{i,j}^2, i = 1, \ldots, I)$. For $(\mu_0^{(j)}, \Sigma_0^{(j)})$, the MLE can be analytically obtained as

$$\hat{\mu}_0^{(j)} = \sum_{i=1}^I \frac{\hat{\beta}_{i,j}}{\hat{\sigma}_{i,j}^2}$$

$$\hat{\Sigma}_0^{(j)} = \frac{1}{I} \sum_{i=1}^I \frac{(\hat{\beta}_{i,j} - \hat{\mu}_0^{(j)})(\hat{\beta}_{i,j} - \hat{\mu}_0^{(j)})^T}{\hat{\sigma}_{i,j}}. \hspace{1cm} (6)$$

The derivation of (6) can be found in [8].

B. Neural Network for Data Fusion and RUL Prediction

In single-degradation-signal based prognostics, the RUL is often defined as the first passage time that the true degradation signal (measurement noise removed) hits a predefined failure threshold. When there are multiple degradation signals, however, it may not be practical to set a failure threshold for each degradation signal. Instead, a HI representing the true underlying degradation process can be constructed through data fusion of all degradation signals, and the RUL can be predicted by estimating the first passage time of the HI.

Suppose $\{\beta_j, \ldots, \beta_J\}$ are the degradation parameters of $J$ sensor signals of a unit. It is assumed that there exists a function $h$ such that $\text{HI}(t) = h(\beta_1, \ldots, \beta_J, t)$. In practical applications, $h$ may be highly nonlinear. Given $\{\beta_1, \ldots, \beta_J\}$, the HI is a deterministic function of time $t$. Therefore, the RUL is a deterministic function of $\{\beta_1, \ldots, \beta_J\}$ and $t$, denoted as $\text{RUL} = g(\beta_1, \ldots, \beta_J, t)$. Due to its excellent function approximation capability, ANNs are proposed to approximate $g$ for data fusion and RUL prediction.

The structure of the proposed NN is shown in Fig. 3. The input layer consists of $J$ sets of degradation parameters $\beta_1, \ldots, \beta_J$ and time $t$. For illustration purposes, only three sets of sensor parameters are shown in Fig. 3 and each sensor signal is modeled by a quadratic polynomial with $\beta_j = (a_j, b_j, c_j)$. Let $p_i$ be the occurred life percentage, i.e., $p_i = t_i / (t_i + \text{RUL}(i))$. We use the $p_i$ as the output layer. The tanh($\cdot$) is selected as the activation function among the input layer and hidden layers due to its obvious advantages over the sigmoid function. Between the last hidden layer and the output layer, the sigmoid activation function is used. In the model training, the sum-of-squared-error is used as the loss function

$$R(\theta) = \sum_{i=1}^I \sum_{k=1}^{K_i} (p_{i,k} - g(\beta_{i,1}, \ldots, \beta_{i,J}, t_k))^2 \hspace{1cm} (7)$$

where $\theta$ is a vector of NN weights or model parameters, $K_i$ is the selected number of time instances of unit $i$ for training, and $p_{i,k}$ is the occurred life percentage at the time $t_k$ for unit $i$.

The NN model is fitted by using the well-known backpropagation algorithm, a generic gradient descent-based approach in NN training. In the training process, all the inputs are standardized to allow one to choose a meaningful range for the random starting weights. Since the loss function is nonconvex and has many local minima, the trained model is quite dependent on the choice of starting weights. Therefore, a large number of random starting weights are tried, and the one with the lowest loss is selected as the final solution.

C. Online Bayesian Updating and RUL Prediction

In the online monitoring and prediction, the degradation parameters of an in-service unit need to be estimated in a real-time manner. With the hyperparameters obtained in Section II-A, it is natural to use the online Bayesian updating scheme to calculate the posterior distribution of all the degradation parameters. Suppose the available sensor data of a unit up to the current time step $k$ is $s_{j,1:k}, j = 1, \ldots, J$, then the posterior target distributions are $p(\beta_j | s_{j,1:k}), j = 1, \ldots, J$. Due to the usage of conjugate priors, all the posterior distributions can be calculated analytically as follows.
Proposition 1: Suppose the priors $\sigma_j^2 \sim IG(\alpha_1^{(j)}, \alpha_2^{(j)})$, $\beta_j | \sigma_j^2 \sim N(\mu_0^{(j)}, \sigma_j^2 \Sigma_0^{(j)})$, then the posterior distributions can be derived as
\[
\begin{align*}
\left( \sigma_j^2 \mid s_{j,1:k} \right) & \sim IG \left( \alpha_1^{(j)} + \frac{k}{2}, \alpha_2^{(j)} + \frac{H_{j,k}}{2} \right) \\
(\beta_j | s_{j,1:k}) & \sim N \left( \mu_0^{(j)}, \sigma_j^2 \Sigma_0^{(j)} \right) \\
(\beta_j | s_{j,1:k}) & \sim \text{MT} \left( \mu_0^{(j)}, \frac{2\sigma_j^2}{\sigma_1^2} + \frac{H_{j,k}}{\sigma_1^2} \Sigma_0^{(j)}, v_{j,k} \right)
\end{align*}
\]

where
\[
\begin{align*}
v_{j,k} & = 2\sigma_1^{(j)} + k \\
\Sigma_k^{(j)} & = \left( X_{j,k}^T X_{j,k} + (\Sigma_0^{(j)})^{-1} \right)^{-1} \\
T_{j,k} & = \left( \Sigma_0^{(j)} \right)^{-1} \mu_0^{(j)} + X_{j,k}^T s_{j,1:k} \\
\mu_k^{(j)} & = \Sigma_k^{(j)} T_{j,k} \\
H_{j,k} & = \left\| s_{j,1:k} \right\|^2 + \left( \mu_0^{(j)} \right)^T (\Sigma_0^{(j)})^{-1} \mu_0^{(j)} - T_{j,k} \Sigma_k^{(j)} T_{j,k} 
\end{align*}
\]

For $\sigma_j^2$ deterministic, i.e., all the units share the same noise variance, which is often an assumption in some existing work and is estimated from historical data [10], [28]. Equation (9) can be used to update the posterior distribution of $\beta_j$. On the other hand, if $\sigma_j^2$ is random and unknown, (10) would be more accurate to capture the uncertainty of $\beta_j$, though its posterior mean is unchanged compared with (9). In practical applications, however, treating $\sigma_j^2$ deterministic or random will influence the hyperparameter estimation, which will result in different $\mu_k^{(j)}$.

In the RUL prediction, the posterior mean or the maximum-a posteriori-probability (MAP) estimate of $\beta_j$ can be plugged into the NN model to get the point estimate of $p_t$. Once $p_t$ is estimated, we can easily obtain the RUL as
\[
\text{RUL}(t) = \frac{1 - p_t}{p_t} t. \quad (11)
\]

Within the Bayesian framework, we can also conveniently obtain the posterior predictive distribution of $p_t$ or the RUL. Specifically, the Monte Carlo approach can be used to generate random samples from the posterior distribution of $\beta_j$, and then these samples are substituted into the NN model to get the predictive samples of RUL.

III. APPLICATION ON C-MAPSS DATA SET

A. Overview of the System and Data Set

In this section, the proposed JPPM is illustrated and evaluated through the benchmarking C-MAPSS data set [25]. C-MAPSS is a tool developed by NASA for simulating a realistic large commercial turbofan engine that is monitored by multiple sensors. The C-MAPSS data set generated in [25] has been widely used as a benchmark system with multiple degradation signals in the prognostics and health management (PHM) field.

C-MAPSS simulates an engine model of the 90,000-lb thrust with altitudes ranging from sea level to 40,000 ft, Mach numbers from 0 to 0.90, and sea-level temperatures from $-60 \, ^\circ\text{F}$ to $103 \, ^\circ\text{F}$. Users can adjust the conditions of aircraft altitude, Mach number, and throttle resolver angle to simulate different environmental conditions [25]. There are 14 inputs to simulate various degradation scenarios. The outputs include various sensor response surfaces and operability margins. A total of 21 variables out of 58 different outputs available from the model are used for analysis, as shown in Table I.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>T2</td>
<td>Total temperature at fan inlet</td>
<td>°R</td>
</tr>
<tr>
<td>T24</td>
<td>Total temperature at LPC</td>
<td>°R</td>
</tr>
<tr>
<td>T30</td>
<td>Total temperature at HPC</td>
<td>°R</td>
</tr>
<tr>
<td>T50</td>
<td>Total temperature at LPT</td>
<td>°R</td>
</tr>
<tr>
<td>P2</td>
<td>Pressure at fan inlet</td>
<td>psia</td>
</tr>
<tr>
<td>P15</td>
<td>Total pressure in bypass-duct</td>
<td>psia</td>
</tr>
<tr>
<td>P30</td>
<td>Total pressure at HPC outlet</td>
<td>psia</td>
</tr>
<tr>
<td>Nf</td>
<td>Physical fan speed</td>
<td>rpm</td>
</tr>
<tr>
<td>Ne</td>
<td>Physical core speed</td>
<td>rpm</td>
</tr>
<tr>
<td>epr</td>
<td>Engine pressure</td>
<td>--</td>
</tr>
<tr>
<td>Ps30</td>
<td>Static pressure at HPC outlet</td>
<td>psia</td>
</tr>
<tr>
<td>phi</td>
<td>Ratio of fuel flow to Ps30</td>
<td>pps/psi</td>
</tr>
<tr>
<td>NRF</td>
<td>Corrected fan speed</td>
<td>rpm</td>
</tr>
<tr>
<td>NRc</td>
<td>Corrected core speed</td>
<td>rpm</td>
</tr>
<tr>
<td>BPR</td>
<td>Bypass Ratio</td>
<td>--</td>
</tr>
<tr>
<td>farB</td>
<td>Burner fuel-air ratio</td>
<td>--</td>
</tr>
<tr>
<td>hBleed</td>
<td>Bleed Enthalpy</td>
<td>--</td>
</tr>
<tr>
<td>NF_dmd</td>
<td>Demanded fan speed</td>
<td>rpm</td>
</tr>
<tr>
<td>PCNIR_d</td>
<td>Demanded corrected fan</td>
<td>rpm</td>
</tr>
<tr>
<td>W31</td>
<td>HPT coolant bleed</td>
<td>lbm/s</td>
</tr>
<tr>
<td>W32</td>
<td>LPT coolant bleed</td>
<td>lbm/s</td>
</tr>
</tbody>
</table>

Fig. 4. Simplified engine diagram simulated in C-MAPSS [25].
A failure threshold for a hidden HI that is not accessible to users is predefined, beyond which the unit is considered failed. A total of four data sets with the corresponding failure modes and operational conditions were generated. In this article, we only consider two of the four data sets, FD001 and FD003, which are commonly used in performance evaluation and comparison. The FD001 data set has a single-failure mode (HPC degradation) and a single-operating condition, while the FD003 data set has one operating condition but two fault modes (HPC and fan degradation). For FD002 and FD004, there are six operating conditions mixed together, all of which affect the sensor values and the degradation process. It is inappropriate to estimate RUL based only on the 21 sensor data. Therefore, these two data sets are not considered here.

For each data set considered in this article, there are 100 training units and 100 testing units. In the training data set, the fault grows in magnitude until system failure. In the test data set, the time series ends sometime prior to system failure. A file of the actual remaining lifetime of the 100 testing units is also included for each data set. Sensor readings from the 21 outputs are collected at each observation epoch for each unit. The prognostic model is developed based on the available degradation patterns of the 100 training units and the testing data set is used for performance evaluation.

### B. Variable Selection and Data Preprocessing

Among these 21 outputs, 14 outputs are highly related to the degradation process with an increasing or decreasing trend, while the other outputs are almost unchanged. Therefore only these 14 degradation signals are included for further selection. The correlation analysis shows that there exist high correlations among these outputs (up to 0.96 for certain pairs). Signals with low correlation exhibit different signal patterns and involve different characteristics of the same unit. Therefore, the outputs are selected in such a way that the data show an obvious degradation trend and the pair-wise correlations of the selected outputs are as low as possible. To select the outputs based on the correlation, the hierarchical clustering algorithm is used, where \((1 - \text{correlation coefficient})\) is used as the distance or dissimilarity measure. The clustering dendrogram is shown in Fig. 5, where five clusters are obtained with a correlation coefficient of 0.75. For each cluster, we randomly select an output, and the final outputs selected for prediction are Nc, T24, BPR, htBleed, and T30.

The typical degradation forms of the selected sensor signals are shown in Fig. 6. All of these signals show an exponential functional form, which has been widely used to model cumulative damage processes [24], [28], [29]. Therefore, we use the exponential function to describe the degradation process of the turbofan engine. Following Liu et al. [1], we first perform log-transformation to the data and then apply linear models to the log-transformed data. Specifically, the quadratic polynomial function is assumed for the log-transformed data of each selected variable.

### C. Results

As discussed in Section II-B, the proposed NN model uses fit parameters and time \(t\) as inputs and the occurred life percentage as output. There are five selected signals, with each having three parameters. Therefore, there are in total of 16 input neurons in the NN model. To reduce the computational cost and to balance the sample size among different units, ten equally spaced time epochs across the whole life span are selected for each unit in the training process. The fivefold cross validation is used on the training data set with 100 units for model structure selection. The model with minimum validation error is eventually selected. It has three layers with six neurons in the first hidden layer and three neurons in the second hidden layer. After the structure is determined, we use the entire training data set to train the model parameters. The MATLAB NN training package is used for training. In the training process, we divide the entire training data set into two parts, including the training part (70%) and the validating part (30%). The training iteration terminates when validating error starts to increase. As the NN is inherently stochastic, \(1 \times 10^4\) sets of initial weights and biases are randomly generated for the iteration to start with, and the trained model with the lowest validation error is used. Once the model is trained, it is applied to the 100 testing units for performance evaluation.

The estimated hyperparameters based on (5) and (6) for the five selected signals are shown in Table II. As we can see from the inverse Gamma parameters \(a_1\) and \(a_2\), BRP has the largest while T24 has the lowest mean and variance for the noise variance \(\sigma^2\) among these five signals. Fig. 7 shows the Bayesian updating process for T30 and htBleed of a randomly selected engine at different time steps. Unsurprisingly,
TABLE II
ESTIMATED HYPERPARAMETERS FOR THE FIVE SELECTED SIGNALS

<table>
<thead>
<tr>
<th>Signal</th>
<th>$\mu_0$</th>
<th>$\Sigma_0$</th>
<th>$(\alpha_1, \alpha_2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T24</td>
<td>$(3.7, -9.8 \times 10^{-5}, 1.2 \times 10^{-6})^T$</td>
<td>$[0.5, 9.0 \times 10^{-4}, -7.0 \times 10^{-6}]$</td>
<td>$(91.8, 0.005)$</td>
</tr>
<tr>
<td>T30</td>
<td>$(3.6, -9.5 \times 10^{-4}, 1.3 \times 10^{-5})^T$</td>
<td>$[-7.0 \times 10^{-6}, -3.9 \times 10^{-7}, 6.0 \times 10^{-9}]$</td>
<td>$(41.3, 0.42)$</td>
</tr>
<tr>
<td>Nc</td>
<td>$(4.1, -1.2 \times 10^{-3}, 1.5 \times 10^{-5})^T$</td>
<td>$[0.4, 2.1 \times 10^{-4}, 4.6 \times 10^{-5}]$</td>
<td>$(11.4, 0.06)$</td>
</tr>
<tr>
<td>BPR</td>
<td>$(-2.3, -1.4 \times 10^{-3}, 2.5 \times 10^{-5})^T$</td>
<td>$[-1.4 \times 10^{-5}, -3.5 \times 10^{-7}, 5.3 \times 10^{-9}]$</td>
<td>$(8.2, 0.26)$</td>
</tr>
<tr>
<td>htBleed</td>
<td>$(2.5, -7.9 \times 10^{-4}, 1.1 \times 10^{-5})^T$</td>
<td>$[-5.2 \times 10^{-4}, 4.5 \times 10^{-5}, -2.8 \times 10^{-7}]$</td>
<td>$(61.8, 0.33)$</td>
</tr>
</tbody>
</table>

The posterior distributions become closer and closer to the true value in terms of the mean and variance when more observations are collected. Note that the posterior distributions for a noise variance $\sigma^2$ also have a similar trend. However, since $\sigma^2$ is not used as input in the NN model, we do not show it here.

In the RUL prediction, two approaches could be used. The first one is to use the posterior means as the point estimates (MAP) of $\beta$ and then substitute them to the NN model to get the point estimate of life percentage $p_t$. The advantage of this approach is its simplicity and low computational cost. Another alternative approach is to calculate the posterior predictive distribution of $p_t$ with the following density function:

$$f(p_t | s_1, s_2, \ldots, s_J, \beta_1, \ldots, \beta_J, \sigma^2) = f(g(t_k, \beta_1, \ldots, \beta_J) | s_1, s_2, \ldots, s_J, \beta_1, \ldots, \beta_J, \sigma^2)$$

where $g(\cdot)$ is the life percentage prediction function based on the proposed NN model. The advantage of this approach is that it makes full use of the uncertainty of $\beta_j, j = 1, \ldots, J$ to produce a predictive distribution instead of a point estimate for life percentage. Due to the high complexity of function $g(\cdot)$, the posterior predictive density function is not tractable analytically. Naturally, the Monte Carlo method is a practical method to approximate the posterior distribution of $p_t$. The posterior predictive mean of $p_t$ is calculated by:

$$\hat{p}_t = \frac{1}{M} \sum_{m=1}^{M} g(t_k, \beta_{1m}, \ldots, \beta_{Jm})$$

where $M$ is the Monte Carlo sample size. One Monte Carlo sample $\beta_{jm}$ is drawn from the posterior distribution of each $\beta_j$ as follows:

$$\beta_{jm} \sim p(\beta_j | s_1, s_2, \ldots, s_J)$$

where $p(\cdot)$ is the prior distribution of $\beta_j$. The reason to use the prior distribution is that the prior distribution is not always available as a function form in the NN model.
Carlo approach could be used, where random samples are drawn from the posterior distributions of $\beta_j$ first, and then these samples are plugged into the NN model to obtain life percentage samples that approximately follow the posterior predictive distribution. In this approach, we can also use the mean of the posterior predictive distribution as a point estimate for $p_t$. In our application study, there is no significant difference between the point estimates using both approaches.

Fig. 8 shows the posterior predictive distributions of life percentage for six randomly selected engines from the testing data set. The prediction is performed at the time of the last observation. Clearly, the prediction is very accurate with mean or MAP close to the true values, even if the prediction is performed at its early stage.

Fig. 9 shows the performance comparison of the JPM with the other two commonly used methods, i.e., the NN model with time $t$ and observation at $t$ as inputs (denoted by NN1), and the one with time $t$, and two consecutive observations as inputs (denoted by NN2) using the $\alpha - \lambda$ metric [30] on six randomly selected engines. The selected structures of NN1 and NN2 are 6-4-2-1 (e.g., six nodes in the input layer and four nodes in the second layer) and 12-8-4-1, respectively. In this metric, $\alpha$ specifies the error bound (here, $\alpha = 15\%$) on the estimated residual life percentage, i.e., $(1 - \alpha)(1 - p_t) \leq (1 - \hat{p}_t) \leq (1 + \alpha)(1 - p_t)$, and $\lambda$ specifies the relative...
distance in time between the prediction point and the actual failure time. It can be seen that almost all the JPM estimated residual life percentages lie in the ±15% error bound, and the prediction becomes more and more accurate as the engine approaches failure. In contrast, the NN1 and NN2 methods have much lower prediction accuracy in most cases and the prediction accuracy is not stable across different time steps.

To further evaluate the performance of JPM and compare it with other methods, we also use the mean value of the absolute percentage error used in [1] as the performance metric, which is defined as

$$\text{err} = \frac{1}{N} \sum_{i=1}^{N} \frac{|\hat{R}_i - R_i|}{T_i}$$

where $\hat{R}_i$ is the predicted residual life at the time of the last measurement for unit $i$ in the testing data set, $R_i$ is the true residual life, $T_i$ is the total life from the beginning to the failure time, and $N$ is the total number of testing units. Fig. 10 shows the mean absolute percentage error of prediction at different levels of an actual remaining lifetime for the proposed JPM with all the five selected signals and with only one of them, respectively. The level label “all” refers to all the testing units, while “T-L” means the testing units with residual life less than or equal to L. As we can see, the proposed JPM by fusing all the five selected signals has much lower prediction error than all the other ones with only one degradation signal.

Fig. 11 shows the comparison of JPM with NN1, NN2, HI_2013 [1], HI_kernel [11], and the LSTM method [31] in terms of the mean absolute percentage error on the testing data set of the FD001 data set. LSTM is a special kind of RNN, and it is well-suited to classifying and making predictions based on time series data. Since it has been recently used for prognostics, we include it here as a state-of-the-art method to compare with. We set the size of the time sliding window to 50, which means 50 continuous measurements of 5 selected signals were used as an input. The tuned LSTM network structure has five hidden layers with 1024, 512, 256, 128 and 64 nodes respectively. The activation function of the middle layers was set to “relu” while that of the output layer was set to sigmoid. Because of the large data size, we use minibatch and batch size was set as 1024. The dropout (the dropout rate is set as 0.2) and fivefold cross validation was used to avoid overfitting.

As shown in Fig. 11, a decreasing trend is observed for almost all these methods, and the proposed JPM outperforms all the other methods significantly in most cases, especially at the early stage. When the units approach failure, more observations will be obtained to infer the degradation trajectory, thus resulting in more accurate predictions. Expectedly, NN2 has higher prediction accuracy than NN1. Due to unsupervised learning, all the HI-based approaches have poor performance at the early prediction stages. For the LSTM approach, the performance is not satisfactory at the early stage. As we explained in Section I, the degradation of all engines is not noticeable in the early life cycles. The input features do not change much while the output decreases linearly. Therefore, it is hard for LSTM to achieve high performance across all the life cycles. Due to the usage of the Bayesian framework and the inclusion of time epoch $\tau$ as the input, the proposed JPM can effectively overcome this issue and achieves excellent performance in both the early and late-life cycles. To further validate the effectiveness of our proposed method, we applied the JPM on another engine data set FD003. Different from the first data set FD001 with only a single-failure mode, the FD003 data set has two fault modes. In the existing literature, few research works investigated the HI-based approaches using the FD003 data set, and therefore, we only compare the JPM with NN1, NN2, and LSTM here. Fig. 12 shows the comparison results, which also shows the effectiveness of the proposed method.
The availability of multisensor data provides us a great opportunity to better monitor the degradation process and improve prognostic accuracy. The existing linear data fusion techniques are incapable of capturing the possible hidden nonlinear relationship, while the traditional NN-based prediction methods fail to fully utilize the past observations before the prediction epoch. To tackle these problems, a JPM was proposed in this article for multisensor data fusion and RUL prediction. In this approach, a Bayesian linear model was used to model each sensor data that partially captures the degradation process, and an ANN was proposed to model the potentially nonlinear relationship between the residual life and multisensor degradation data. At the offline stage, an empirical two-stage process was used to estimate the hyperparameters of prior distributions. At the online stage, a Bayesian updating scheme was used to update the posterior distribution of model parameters of an in-service unit, and the updated parameters were used as input of the NN for residual life prediction. The developed method was demonstrated and validated using the C-MAPSS data set. The performance and comparison results had shown that JPM has much higher prediction accuracy than the existing approaches at most of the prediction epochs.

The advantages of the proposed JPM are twofold: 1) by using the Bayesian filtered parameters and epoch $t$ as input of the NN model, the JPM can fully utilize the multisensor data before the prediction epoch to capture the degradation trajectory and 2) compared with RNN, LSTM or deep CNN approaches, the proposed JPM has a simple network structure and relatively excellent performance in the early life cycles without clear degradation trend. It is worth noting that the proposed JMP is developed for the degradation process under the same operating condition. When multiple operating conditions are mixed together, however, there may not be clear degradation trends for the multisensor data, making the JPM not applicable. The extension to multiple operating conditions will be investigated in the future.


**REFERENCES**


Yuanyuan Gao received the B.S. degree in aircraft design and engineering from Beihang University, Beijing, China, in 2018. She is currently pursuing the Ph.D. degree in industrial engineering and management with Peking University, Beijing.

Her research interests are focused on data mining, advanced data analytics, quality, and reliability engineering.

Yuxin Wen received the B.S. degree in medical informatics engineering from Sichuan University, Chengdu, China, in 2011, the M.S. degree in biomedical engineering from Zhejiang University, Zhejiang, China, in 2014. She is currently pursuing the Ph.D. degree in electrical and computer engineering with The University of Texas at El Paso, El Paso, TX, USA.

Her research interests are focused on statistical modeling, prognostics, and reliability analysis.

Jianguo Wu (Member, IEEE) received the B.S. degree in mechanical engineering from Tsinghua University, Beijing, China, in 2009, the M.S. degree in mechanical engineering from Purdue University, West Lafayette, IN, USA, in 2011, and the M.S. degree in statistics and the Ph.D. degree in industrial and systems engineering from the University of Wisconsin–Madison, Madison, WI, USA, in 2014 and 2015, respectively.

He was an Assistant Professor with the Department of Industrial, Manufacturing and Systems Engineering, The University of Texas at El Paso, El Paso, TX, USA, from 2015 to 2017. He is currently an Assistant Professor with the Department of Industrial Engineering and Management, Peking University, Beijing, China. His research interests are focused on data-driven modeling, monitoring, and analysis of advanced manufacturing processes and complex systems for quality control and reliability improvement.

Dr. Wu is a member of the Institute for Operations Research and Management Science (INFORMS), the Institute of Industrial and Systems Engineers (IISE), and the Society of Manufacturing Engineers (SME).