Model-based Prognostics under Limited Sensing[1]

Matthew Daigle[1] and Kai Goebel[2]

[1]University of California, Santa Cruz, NASA Ames Research Center, Moffett Field, CA, USA

[2]NASA Ames Research Center, Moffett Field, CA, USA

{matthew.j.daigle,kai.goebel}@nasa.gov

Abstract—Prognostics is crucial to providing reliable condition-based maintenance decisions. To obtain accurate predictions of component life, a variety of sensors are often needed. However, it is typically difficult to add enough sensors for reliable prognosis, due to system constraints such as cost and weight. Model-based prognostics helps to offset this problem by exploiting domain knowledge about the system, its components, and how they fail by casting the underlying physical phenomena in a physics-based model that is derived from first principles. We develop a model-based prognostics methodology using particle filters, and investigate the benefits of a model-based approach when sensor sets are diminished. We apply our approach to a detailed physics-based model of a pneumatic valve, and perform comprehensive simulation experiments to demonstrate the robustness of model-based approaches under limited sensing scenarios using prognostics performance metrics.

TABLE OF CONTENTS

1. INTRODUCTION

Prognostics is a key enabling technology for applying condition-based maintenance. The goal of prognostics is to make *end of life* (EOL) and *remaining useful life* (RUL) predictions that enable timely maintenance decisions to be made. As with diagnostics, prognostics methods may typically be categorized as either data-driven or model-based. Data-driven approaches do not take advantage of system and domain knowledge, but instead use learning methods to identify trends and determine EOL and RUL [1] Such methods rely on large amounts of run-to-failure data that are used to train the algorithms. The training becomes dependent on the set of sensors used, and, therefore, is not robust to loss of sensors or a possible lack of sensors. Furthermore, the required training data, in many cases, does not exist. In contrast, model-based approaches exploit domain knowledge of the system, its components, and how they

fail in order to provide EOL and RUL predictions [2]-[5]. The underlying physical phenomena are captured in a physics-based model that is derived from first principles, therefore, model-based approaches can provide EOL and RUL estimates that are much more accurate and precise than data-driven approaches, if the models are correct. In addition, model-based approaches can be robust to sensor loss and still work under limited sensing environments with an accurate model.

In this paper, we demonstrate the effectiveness of model-based approaches under limited sensing conditions. Reduction of the number of sensors implies an increase in uncertainty. Other sources of uncertainty are also present, such as model inaccuracies, sensor noise, and uncertainty in future input profiles. To help manage this uncertainty, we apply a particle filtering approach that is based on joint state-parameter estimation. Particle filters approximate the posterior probability distribution as a set of discrete, weighted samples. Although suboptimal, the advantage of particle filters is that they can be applied to systems which may be nonlinear and have non-Gaussian noise terms, where optimal solutions are unavailable or intractable. Particle filtering approaches have previously been successful in prognostics applications. In [4], the authors apply a particle filtering approach to prediction of end of discharge and EOL in lithium-ion batteries, which highlights the performance that can be achieved by having accurate models. In [6], the authors apply a particle filter-based prognosis method to prediction of battery grid corrosion. Both [7] and [8] apply particle filtering to estimation and prediction of crack growth. Performance improvements over standard particle filter methods can be achieved in prognostics with advanced filtering techniques, such as the use of correction loops [9], fixed-lag filters [5], Rao-Blackwellized particle filters [10],[11], and risk-sensitive particle filters [12].

In this paper, we develop a model-based prognostics framework that incorporates particle filters. As a case study, we construct a detailed physics-based model of a pneumatic valve that includes models of different damage mechanisms. We run a comprehensive set of prognostics experiments in simulation to demonstrate the robustness of the approach to limited sensor sets, evaluated using established prognostic metrics [13], [14].
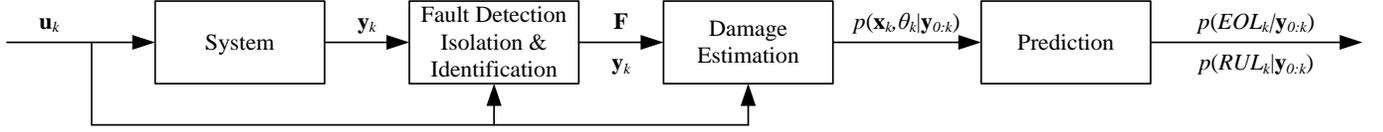

Figure 1 - Prognostics architecture.

The paper is organized as follows. Section 2 formulates the prognostics problem and overviews the computational architecture we adopt. Section 3 develops the damage estimation method using particle filters. Section 4 describes the prediction procedure. Section 5 presents the pneumatic valve case study with experimental results in simulation. Section 6 concludes the paper.

2. PROGNOSTICS APPROACH

Problem Formulation

The problem of prognostics is to predict the EOL and/or the RUL of a component. In this paper, we develop a general model-based approach, where the system model is given by

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{t}, \mathbf{x}(t), \theta(t), u(t), v(t))$$
$$y(t) = h(t, x(t), \theta(t), u(t), n(t))$$

where $x(t) \in R^{n_x}$ is the state vector, $\theta(t) \in R^{n_\theta}$ is the parameter vector, $\mathbf{u}(t) \in R^{n_u}$ is the input vector, $\mathbf{v}(t) \in R^{n_v}$ is the process noise vector, $\mathbf{f}$ is the state equation, $y(t) \in R^{n_y}$ is the output vector, $n(t) \in R^{n_n}$ is the measurement noise vector, and $\mathbf{h}$ is the output equation.

Our goal is to predict EOL at a given time point t_P using the discrete sequence of observations up to time t_P, denoted as $\mathbf{y}_{0:t_P}$. In order to determine when EOL has been reached, we require a condition that is a function of the system state and parameters, $C_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t))$, which determines whether EOL has been reached, where

$$C_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t)) = \begin{cases} 1, & \text{if EOL is reached} \\ 0, & \text{otherwise}. \end{cases}$$

Using this function, we can define EOL with

$$EOL(t_P) \equiv \arg\min_{t \geq t_P} C_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t)) = 1,$$

and RUL with

$$RUL(t_P) \equiv EOL(t_P) - t_P.$$

Because of the noise inherent in the process, the measurements, and future inputs, we must compute a probability distribution of the EOL or RUL, i.e., the goal is to compute, at time t_P, $p(EOL(t_P) | \mathbf{y}_{0:t_P})$ or $p(RUL(t_P) | \mathbf{y}_{0:t_P})$.

Prognostics Architecture

We adopt a model-based approach, wherein we develop detailed physics-based models of components and systems that include descriptions of how fault parameters evolve in time. These models depend on unknown and possibly time-varying wear parameters, $\boldsymbol{\theta}(t)$. Therefore, our solution to the prognostics problem takes the perspective of joint state-parameter estimation. In discrete time k, we estimate $\mathbf{x}_k$ and $\boldsymbol{\theta}_k$, and use these estimates to predict EOL and RUL at desired time points.

We employ the prognostics architecture in Fig. 1. The system is provided with inputs $\mathbf{u}_k$ and provides measured outputs $\mathbf{y}_k$. The fault detection, isolation, and identification (FDII) module provides a fault set $\mathbf{F}$, which is used by the damage estimation module to determine estimates of the states and unknown parameters, represented as a probability distribution $p(\mathbf{x}_k, \boldsymbol{\theta}_k | \mathbf{y}_{0:k})$. This distribution is used by the prediction module which computes EOL and RUL using hypothesized future inputs. EOL and RUL are computed as probability distributions $p(EOL_{k_P} | \mathbf{y}_{0:k_P})$ and $p(RUL_{k_P} | \mathbf{y}_{0:k_P})$. In this paper, we focus on the damage estimation and prediction modules, and assume a solution to FDII.

3. DAMAGE ESTIMATION

To estimate the damage, we need to estimate $p(\mathbf{x}_k, \boldsymbol{\theta}_k | \mathbf{y}_{0:k})$. In this paper, we use the particle filter for this purpose [15],[16]. With particle filters, the state distribution is approximated by a set of discrete weighted samples, or particles, $\{(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i), w_k^i\}_{i=1}^N$, where N denotes the number of particles, and for particle i, $\mathbf{x}_k^i$ denotes the state estimates, $\boldsymbol{\theta}_k^i$ denotes the parameter estimates, and w_k^i denotes the weight.

Particle filters are best suited to estimation in nonlinear systems with possibly non-Gaussian noise, where optimal solutions are unavailable or intractable. In this respect, they may be viewed as a general (suboptimal) solution to

Algorithm 1 SIR Filter

Inputs: $\{(\mathbf{x}_{k-1}^i, \boldsymbol{\theta}_{k-1}^i), w_{k-1}^i\}_{i=1}^N, \mathbf{u}_{k-1:k}, \mathbf{y}_k$
Outputs: $\{(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i), w_k^i\}_{i=1}^N$
for $i = 1$ **to** N **do**
   $\boldsymbol{\theta}_k^i \sim p(\boldsymbol{\theta}_k | \boldsymbol{\theta}_{k-1}^i)$
   $\mathbf{x}_k^i \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}^i, \boldsymbol{\theta}_{k-1}^i, \mathbf{u}_{k-1})$
   $w_k^i \leftarrow p(\mathbf{y}_k | \mathbf{x}_k^i, \boldsymbol{\theta}_k^i, \mathbf{u}_k)$
end for
$W \leftarrow \sum_{i=1}^N w_k^i$
for $i = 1$ **to** N **do**
   $w_k^i \leftarrow w_k^i / W$
end for
$\{\mathbf{x}_k^i, \boldsymbol{\theta}_k^i, w_k^i\}_{i=1}^N \leftarrow \texttt{Resample}(\{\mathbf{x}_k^i, \boldsymbol{\theta}_k^i, w_k^i\}_{i=1}^N)$

Algorithm 2 EOL Prediction

Inputs: $\{(\mathbf{x}_{k_P}^i, \boldsymbol{\theta}_k^i), w_{k_P}^i\}_{i=1}^N$
Outputs: $\{EOL_{k_P}^i, w_{k_P}^i\}_{i=1}^N$
for $i = 1$ **to** N **do**
   $k \leftarrow k_P$
   $\mathbf{x}_k^i \leftarrow \mathbf{x}_{k_P}^i$
   $\boldsymbol{\theta}_k^i \leftarrow \boldsymbol{\theta}_{k_P}^i$
   while $C_{EOL}(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i) = 0$ **do**
     Predict $\hat{\mathbf{u}}_k$
     $\boldsymbol{\theta}_{k+1}^i \sim p(\boldsymbol{\theta}_{k+1} | \boldsymbol{\theta}_k^i)$
     $\mathbf{x}_{k+1}^i \sim p(\mathbf{x}_{k+1} | \mathbf{x}_k^i, \boldsymbol{\theta}_k^i, \hat{\mathbf{u}}_k)$
     $k \leftarrow k + 1$
     $\mathbf{x}_k^i \leftarrow \mathbf{x}_{k+1}^i$
     $\boldsymbol{\theta}_k^i \leftarrow \boldsymbol{\theta}_{k+1}^i$
   end while
   $EOL_{k_P}^i \leftarrow k$
end for

---

the state estimation problem. Performance can be improved by increasing the number of particles, but this also results in higher computational costs. The number of particles must be chosen to suit the application requirements.

We employ the sampling importance resampling (SIR) particle filter, and implement the resampling step using systematic resampling [17]. In particle filters, the posterior density is approximated by

$$p(\mathbf{x}_k, \boldsymbol{\theta}_k \mid \mathbf{y}_{0:k}) \approx \sum_{i=1}^N w_k^i \delta_{(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i)}(d\mathbf{x}_k d\boldsymbol{\theta}_k),$$

where $\delta_{(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i)}(d\mathbf{x}_k d\boldsymbol{\theta}_k)$ denotes the Dirac delta function located at $(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i)$.

Here, the parameters $\boldsymbol{\theta}_k$ evolve by some unknown random process that is independent of the state $\mathbf{x}_k$. To perform parameter estimation within a particle filter framework, however, we need to assign some type of evolution to the parameters. The typical solution is to use a random walk, i.e., for parameter θ, $\theta_k = \theta_{k-1} + \xi_{k-1}$, where ξ_{k-1} is typically Gaussian noise. With this type of evolution, during the sampling step particles are generated with parameter values that will be different from the initial guesses for the unknown parameters. The particles with parameter values closest to the true values should be assigned higher weight, thus allowing the particle filter to converge to the true values. The selected variance of the random walk noise must be large enough so as to allow convergence in a reasonable amount of time, but small enough such that when convergence is reached, the parameter can be tracked smoothly. Since the parameter values are unknown to start with, this can be a difficult task, but knowledge of the correct order of magnitude of the parameter is helpful. Additionally, correction loop methods can be used to tune this value online as a function of performance [9].

The pseudocode for a single step of the SIR filter is shown as Algorithm 1. Each particle is propagated forward to time k by first sampling new parameter values and sampling new states. The particle weight is assigned using $\mathbf{y}_k$. The weights are then normalized, followed by the resampling step (see [15] for pseudocode of the systematic resampling algorithm).

4. PREDICTION

In the prediction phase, we wish to compute at time k_P, $p(EOL_{k_P} \mid \mathbf{y}_{0:k_P})$ and $p(RUL_{k_P} \mid \mathbf{y}_{0:k_P})$. The particle filter computes

$$p(\mathbf{x}_{k_P}, \boldsymbol{\theta}_{k_P} \mid \mathbf{y}_{0:k_P}) \approx \sum_{i=1}^N w_{k_P}^i \delta_{(\mathbf{x}_{k_P}^i, \boldsymbol{\theta}_{k_P}^i)}(d\mathbf{x}_{k_P} d\boldsymbol{\theta}_{k_P}).$$

We can approximate a prediction distribution n steps forward as [18]

$$p(\mathbf{x}_{k_P+n}, \boldsymbol{\theta}_{k_P+n} \mid \mathbf{y}_{0:k_P}) \approx$$
$$\sum_{i=1}^N w_{k_P}^i \delta_{(\mathbf{x}_{k_P+n}^i, \boldsymbol{\theta}_{k_P+n}^i)}(d\mathbf{x}_{k_P+n} d\boldsymbol{\theta}_{k_P+n}).$$

So, for a particle i propagated n steps forward (without new data), we can simply take its weight as $w_{k_P}^i$. Similarly, we can approximate the EOL as

$$p(EOL_{k_P} \mid \mathbf{y}_{0:k_P}) \approx \sum_{i=1}^N w_{k_P}^i \delta_{(EOL_{k_P}^i)}(dEOL_{k_P}).$$

To compute EOL, then, we propagate each particle forward to its own EOL and use that particle's weight at k_P for the weight of its EOL prediction.

The pseudocode for the prediction procedure is given as Algorithm 2. Each particle i is propagated forward until $C_{EOL}(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i)$ evaluates to 1, at this point EOL has been
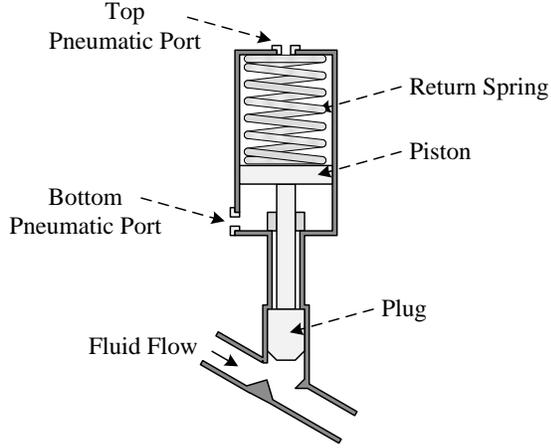
Figure 2 - Pneumatic valve.

reached for this particle. Prediction requires hypothesizing future inputs of the system, $\hat{\mathbf{u}}_k$. The inputs must be chosen carefully because different inputs often have different effects on damage progression. If future inputs are unknown, inputs for each particle may be selected based on a distribution of possible anticipated input values.

5. CASE STUDY

In order to illustrate our prognostics methodology, we take a pneumatic valve as a case study. We develop a physics-based model of the valve and its damage mechanisms. We then present simulation experiments to demonstrate joint state-parameter estimation and EOL/RUL prediction for different sets of sensors and noise levels. The scenarios are compared using prognostics performance metrics established in [13], [14].

Component Modeling

Pneumatic valves are complex mechanical systems used in many domains. These valves are actuated by gas, and can use different types of actuators. A normally-closed valve with a linear cylinder actuator is depicted in Fig. 2. The valve is opened by filling the chamber below the piston with gas up to the supply pressure, and evacuating the chamber above the piston down to atmospheric pressure. The valve is closed by filling the chamber above the piston, and evacuating the chamber below the piston. The return spring ensures that when pressure is lost, the valve will close due to the force exerted by the return spring.

We develop a physics model of the valve based on mass and energy balances. The system state includes the position of the valve, $x(t)$, the velocity of the valve, $v(t)$, the mass of the gas in the volume above the piston, $m_t(t)$,

and the mass of the gas in the volume below the piston, $m_b(t)$:

$$\mathbf{x}(t) = \begin{bmatrix} x(t) \\ v(t) \\ m_t(t) \\ m_b(t) \end{bmatrix}.$$

The position when the valve is fully closed is defined as $x=0$. The stroke length of the valve is denoted by L_s; when the valve is fully open its position is $x=L_s$.

The derivatives of the states are described by

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} v(t) \\ \dfrac{1}{m}\sum F(t) \\ f_t(t) \\ f_b(t) \end{bmatrix},$$

where m is the combined mass of the piston and plug, $\sum F(t)$ is the sum of forces acting on the valve, and $f_t(t)$ and $f_b(t)$ are the mass flows going into the top and bottom pneumatic ports, respectively.

The inputs are considered to be

$$\mathbf{u}(t) = \begin{bmatrix} p_l(t) \\ p_r(t) \\ u_t(t) \\ u_b(t) \end{bmatrix},$$

where $p_l(t)$ and $p_r(t)$ are the fluid pressures on the left and right side of the plug, respectively, and $u_t(t)$ and $u_b(t)$ are the input pressures to the top and bottom pneumatic ports. These pressures will alternate between the supply pressure and atmospheric pressure depending on the commanded valve position.

The sum of forces acting on the piston includes (1) the forces from the pneumatic gas: $(p_b(t)-p_t(t))A_p$, where $p_b(t)$ and $p_t(t)$ are the gas pressures on the bottom and the top of the piston, respectively, and A_p is the surface area of the piston, (2) the forces from the fluid flowing through the valve: $(p_r(t)- p_l(t))A_v$, where A_v is the area of the valve contacting the fluid, (3) the weight of the moving parts of the valve: $-mg$, where g is the acceleration due to gravity, (4) the spring force: $-k(x(t)-x_o)$, where k is the spring constant and x_o is the amount of spring compression when the valve is closed, (5) friction: $-rv(t)$, where r is the coefficient of kinetic friction, and (6) the contact forces at the boundaries of the valve motion:

$$
\begin{cases}
k_c(-x), & x < 0 \\
0, & 0 \le x \le L_s \\
-k_c(x - L_s) & x > L_s,
\end{cases}
$$

where k_c is the (large) spring constant associated with the flexible seals.

The pressures $p_t(t)$ and $p_b(t)$ are calculated as:

$$
p_t(t) = \frac{m_t(t)R_g T}{V_{t_0} + A_p(L_s - x(t))}
$$

$$
p_b(t) = \frac{m_b(t)R_g T}{V_{b_0} + A_p x(t)}
$$

where we assume an isothermal process in which the (ideal) gas temperature is constant at T, R_g is the gas constant for the pneumatic gas, and V_{t_0} and V_{b_0} are the minimum gas volumes for the gas chambers above and below the piston, respectively.

The gas flows are given by:

$$
f_t(t) = f_g(p_t(t), u_t(t))
$$

$$
f_b(t) = f_g(p_b(t), u_b(t))
$$

where f_g defines gas flow through an orifice for choked and non-choked flow conditions [19]:

$$
f_g(p_1, p_2) =
$$

$$
\begin{cases}
C_s A_s p_1 \sqrt{\dfrac{\gamma}{ZR_g T}\left(\dfrac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}, \\
\qquad p_1 \ge p_2 \wedge p_1/p_2 \ge \left(\frac{\gamma+1}{2}\right)^{\gamma/(\gamma-1)} \\[6pt]
C_s A_s p_1 \sqrt{\dfrac{\gamma}{ZR_g T}\left(\dfrac{2}{\gamma-1}\right)\left(\left(\dfrac{p_2}{p_1}\right)^{\frac{2}{\gamma}} - \left(\dfrac{p_2}{p_1}\right)^{\frac{\gamma+1}{\gamma}}\right)}, \\
\qquad p_1 \ge p_2 \wedge p_1/p_2 < \left(\frac{\gamma+1}{2}\right)^{\gamma/(\gamma-1)} \\[6pt]
C_s A_s p_2 \sqrt{\dfrac{\gamma}{ZR_g T}\left(\dfrac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}, \\
\qquad p_1 < p_2 \wedge p_2/p_1 \ge \left(\frac{\gamma+1}{2}\right)^{\gamma/(\gamma-1)} \\[6pt]
C_s A_s p_2 \sqrt{\dfrac{\gamma}{ZR_g T}\left(\dfrac{2}{\gamma-1}\right)\left(\left(\dfrac{p_1}{p_2}\right)^{\frac{2}{\gamma}} - \left(\dfrac{p_1}{p_2}\right)^{\frac{\gamma+1}{\gamma}}\right)}, \\
\qquad p_1 < p_2 \wedge p_2/p_1 < \left(\frac{\gamma+1}{2}\right)^{\gamma/(\gamma-1)}
\end{cases}
$$

where γ is the ratio of specific heats, Z is the gas compressibility factor, C_s is the flow coefficient, and A_s is
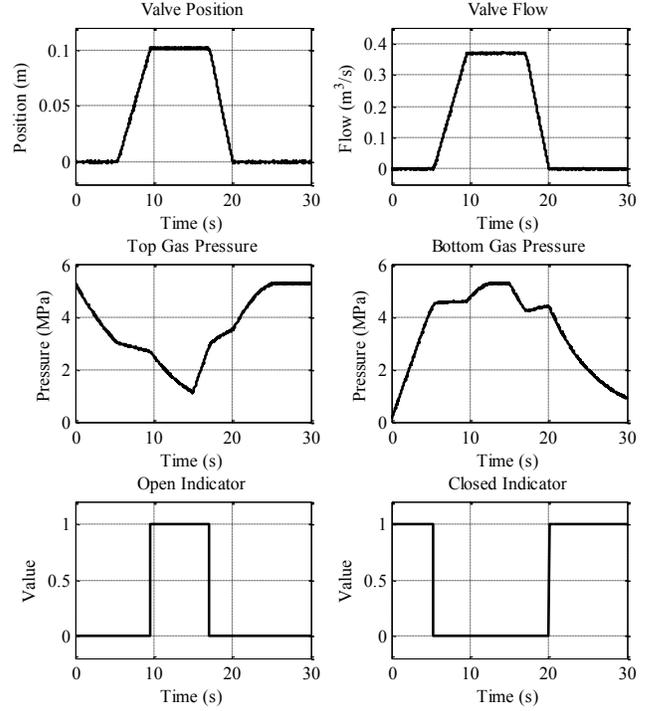

Figure 3 - Nominal valve operation.

the orifice area. Choked flow occurs when the pressure ratio exceeds $\left(\frac{\gamma+1}{2}\right)^{\gamma/(\gamma-1)}$.

We select our complete measurement vector as

$$
\mathbf{y}(t) =
\begin{bmatrix}
x(t) \\
p_t(t) \\
p_b(t) \\
f_v(t) \\
open(t) \\
closed(t)
\end{bmatrix},
$$

where f_v is the fluid flow through the valve:

$$
f_v(t) = \frac{x(t)}{L_s} C_v A_v \sqrt{\frac{2}{\rho}\left|p_{fl} - p_{fr}\right|}\, \text{sign}(p_{fl} - p_{fr}),
$$

C_v is the (dimensionless) flow coefficient of the valve, ρ is the liquid density, and we assume a linear flow characteristic for the valve. The $open(t)$ and $closed(t)$ measurements are discrete sensors which output 1 if the valve is in the fully opened or fully closed state:
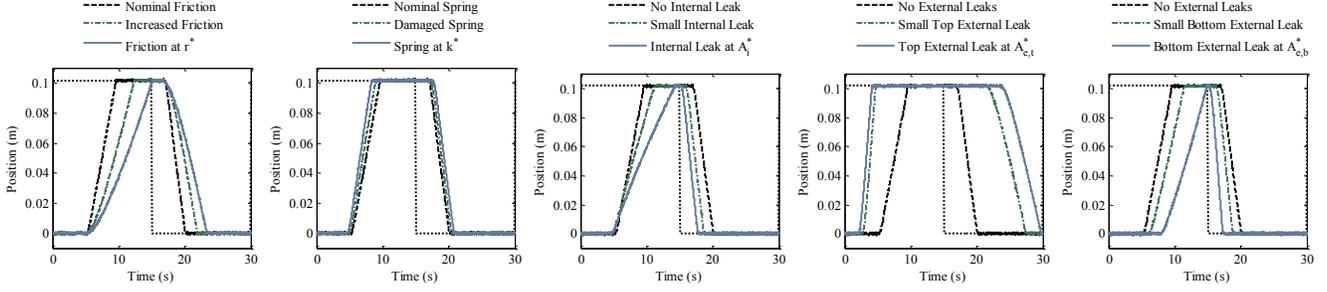
Figure 4 - Valve operation with increasing wear.

$$open(t) = \begin{cases} 1, & x(t) \geq L_s \\ 0, & \text{otherwise} \end{cases}$$

$$closed(t) = \begin{cases} 1, & x(t) \leq 0 \\ 0, & \text{otherwise} \end{cases}$$

It should be noted how the discrete position sensors (*open* and *closed*) are treated within the particle filter framework. In a particle filter, a certain amount of sensor noise must be assumed for sensors, but, in practice, the discrete position sensors have no noise. If an artificial amount of noise is not assumed within the particle filter for these sensors, then sever degeneracy will result, because very few, if any, particles will be able to predict the transitions of these sensor values. The degeneracy is severe enough that it cannot be remedied by resampling methods. Assuming some amount of sensor noise addresses this issue.

Fig. 3 shows a nominal valve cycle. The valve is commanded to open at 0 s. The top pneumatic port opens to atmosphere and the bottom opens to the supply pressure (approximately 5.3 MPa, or 750 psig). When the force on the underside of the piston is large enough to overcome the return spring, friction, and the gas force on the top of the piston, the valve begins to move upward as the pneumatic gas continues to flow into and out of the valve actuator. At about 8 s the valve is completely open. The valve is commanded to close at 15 s. The bottom pneumatic port opens to atmosphere and the bottom opens to the supply pressure. When the force balance becomes negative, the valve starts to move downward, and completely closes at around 20 s. The valve closes faster than it opens due to the return spring.

Damage Modeling

Our general approach to damage modeling is as follows. First, we identify parameters in the model that characterize the extent of specific forms of damage, and these augment the state vector **x**. We then incorporate models of how those parameters change over time with system operation. It is the parameters of these equations that are unknown and must be estimated, and also augment **x** for that purpose. In the valve model, we consider damage or wear characterized by the increase in

friction coefficient, the decrease in spring constant, the appearance and growth of an internal valve leak between the volumes on either side of the piston, and the appearance and growth of external leaks at the pneumatic ports.

One damage mechanism present in valves is sliding wear. The equation for sliding wear takes on the following form [20]:

$$\dot{V}(t) = w|F(t)v(t)|,$$

where $V(t)$ is the wear volume, w is the wear coefficient (which depends on material properties such as hardness), $F(t)$ is the sliding force, and $v(t)$ is the sliding velocity. Friction will increase linearly with sliding wear, because the contact area between the sliding bodies becomes greater as surface asperities wear down [20]. Lubrication between the sliding bodies can also degrade over time. We therefore model the change in friction coefficient in a form similar to sliding wear:

$$\dot{r}(t) = w_r|F_f(t)v(t)|,$$

where w_r is the wear coefficient, and $F_f(t)$ is the friction force defined in the previous subsection. Fig. 4 shows the effect of an increase in friction on the valve cycle. From the simulation, we can determine the value of the friction parameter, r^*, at which the valve has reached EOL. At this value, the friction force becomes large enough that the valve cannot open within the 15 s limit, as shown in Fig. 4. So, $C_{EOL}(\mathbf{x}(t),\boldsymbol{\theta}(t)) = 1$ if $r(t) \geq r^*$.

We assume the same equation form for spring damage:

$$\dot{k}(t) = -w_k|F_s(t)v(t)|,$$

where w_k is the spring wear coefficient and $F_s(t)$ is the spring force. The more the spring is used, the weaker it becomes. We define k^* as the largest value of k at which the valve will not fully close upon loss of supply pressure. Fig. 4 shows the effect of a decrease in the spring parameter on the valve cycle. In normal operation, without
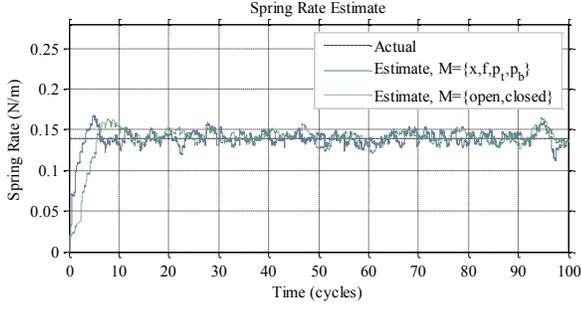
Figure 5 - Estimation results for w_k with $M=\{x,f,p\mathbf{t},p_b\}$ and $M=\{open,closed\}$.

the spring tending the valve to close, the valve will open faster and close slower. However, the spring must be strong enough to close the valve against system pressure when the actuating pressure is lost. So, $C_{EOL}(\mathbf{x}(t),\boldsymbol{\theta}(t))=1$ also if $k(t) \leq k^*$.

An internal leak in the valve can appear at the seal surrounding the piston as a result of sliding wear. The pneumatic gas is then able to flow between the volumes above and below the piston, decreasing the response time of the valve. We parameterize this leak by its equivalent orifice area, $A_i(t)$, described by:

$$\dot{A}_i(t) = w_i \left| F_f(t)v(t) \right|,$$

where w_i is the wear coefficient. The mass flow at the leak, $f_i(t)$, is computed using the gas flow equation:

$$f_i(t) = f_g(p_t(t), p_b(t))$$

As sliding wear occurs, the leak size keeps increasing. The presence of an internal leak makes it more difficult to actuate the valve, because it causes gas to flow into the lower pressure volume that is being evacuated and out of the higher pressure volume that is being filled. We define A_i^* as the minimum internal leak area at which the valve cannot open within the 15 s limit. So, $C_{EOL}(\mathbf{x}(t),\boldsymbol{\theta}(t))=1$ also if $A_i \geq A_i^*$. Fig. 4 shows the effect of an internal leak on the valve cycle.

External leaks can also form, most likely at the actuator connections to the pneumatic gas supply, due to corrosion and other environmental factors. Without knowledge of how the leak size progresses, we assume the growth of the area of the leak holes, $A_e(t)$, is linear:

$$\dot{A}_e(t) = w_e,$$

where w_e is the wear coefficient. We use additional t and b subscripts to denote leaks at the top and bottom pneumatic ports, respectively. The effect of the formation of a leak at

the top pneumatic port is that it becomes easier to open the valve but more difficult to close it. Conversely, the effect of a leak at the bottom pneumatic port is that it becomes more difficult to open but easier to close the valve. Through simulation we can determine the minimum size leak holes at which the valve cannot open or close within the 15 s limit, $A_{e,t}^*$ and $A_{e,b}^*$. (An alternative is to use a maximum allowable leakage rate to define EOL.) So, $C_{EOL}(\mathbf{x}(t),\boldsymbol{\theta}(t))=1$ also if $A_{e,t}(t) \geq A_{e,t}^*$ or $A_{e,b}(t) \geq A_{e,b}^*$. Fig. 4 shows the effect of external leak on the valve cycle.

Experimental Results

We performed a number of simulation experiments to validate our prognostics methodology and evaluate how it performs under different sensor sets using the prognostics performance metrics described in [13], [14].

In each experiment, we considered additive zero-mean process and measurement noise, used N=500 particles, and used a sample time of 0.01 s. We assumed that only a single damage mechanism was active, and, in each experiment, started from the point where damage has been identified and the only unknown is the wear coefficient (initially assumed to be 0 for parameter estimation). The wear coefficients were chosen so that EOL would be reached between 100 and 150 cycles.

The particle filters must be tuned to use the appropriate amount of noise. Assuming the variance of the process was known, the particle filters were set to sample with the same amount of process noise. The random walk variances were selected assuming the orders of magnitude of the wear coefficients were known. The particle filters were also set to assume 10 times the amount of actual measurement noise (assumed to be known). For the discrete position sensors, a value of 0.1 was used for the variance of these sensors, which resulted in the best overall estimation performance for these scenarios. With these settings fixed, we then varied only the measurement set, denoted as M.

First, we provide an example scenario that demonstrates our model-based methodology and explains the metrics used to evaluate performance. We consider the case of spring damage, where w_k=0.14, EOL^*=106.6 cycles (where EOL^* and RUL^* denote the true EOL and RUL), and examine two cases in detail: one with $M=\{x,f,p_t,p_b\}$, and one with $M=\{open,closed\}$. Fig. 5 shows the estimation of the hidden parameter, w_k for these two cases. As observed, the estimates converge then track the true wear parameter value. The two cases appear to have comparable estimation accuracy. Clearly though, the $M=\{open,closed\}$ case is slower to converge, achieving convergence within 15 cycles, whereas the $M=\{x,f,p_t,p_b\}$ converges within 10. We evaluate the performance of the

wear parameter estimation using three metrics, for quantifying accuracy, spread, and convergence.

Accuracy is calculated using the percentage root mean square error (PRMSE), which expresses relative estimation accuracy as a percentage:

$$\text{PRMSE} = 100 \sqrt{\text{Mean}_k \left[\left(\frac{\hat{w}_k - w_k^*}{w_k^*} \right)^2 \right]},$$

where $\hat{w}_k$ denotes the estimated wear parameter value at time k, w_k^* denotes the true wear parameter value at k, and Mean_k denotes the mean over all values of k. In computing PRMSE, we ignore the first 10 cycles (300 s), where most of the error is associated with convergence. In this case, the PRMSE is 5.11% for $M=\{x,f,p_t,p_b\}$, and 5.23% for $M=\{open,closed\}$.

We calculate the spread using the relative median absolute deviation (RMAD), which expresses the spread relative to the median as a percentage:

$$\text{RMAD}(X) = 100 \frac{\text{Median}_i \left(\left| X_i - \text{Median}_j(X_j) \right| \right)}{\text{Median}_j(X_j)}$$

where X is a data set and X_i is an element of that set. For estimation spread, for time k we use the distribution of wear parameter values given by the particle set at k as the data set. Note that since we use the SIR filter, all particles are equally weighted and the median can be directly applied. We denote the average RMAD over multiple k using $\overline{\text{RMAD}}_w$:

$$\overline{\text{RMAD}}_w = \text{Mean}_k(\text{RMAD}_{w,k})$$

where $\text{RMAD}_{w,k}$ denotes the RMAD of the wear parameter at time k. In this example, $\overline{\text{RMAD}}_w=11.00\%$ for $M=\{x,f,p_t,p_b\}$ and $\overline{\text{RMAD}}_w=16.23\%$ for $M=\{open,closed\}$. Although estimation accuracy is comparable, the spread is significantly better when using the full set of continuous sensors.

The final estimation metric is convergence of the wear parameter estimate, denoted as C_w. We use the definition of the convergence metric described in [13], where the convergence of a curve is expressed as the distance from the origin to the centroid under the curve (a shorter distance is better). We use the absolute error of the hidden parameter estimate as the curve. In this case, $C_w=27.99$ for $M=\{x,f,p_t,p_b\}$, and $C_w=44.04$ for $M=\{open,closed\}$. Since the wear parameter is many orders of magnitude smaller than the time scale, the convergence score is approximately in units of seconds. We take convergence only over the first 100 s so that errors after convergence

do not contribute. Otherwise, a curve with fast convergence but poor tracking could score higher than a curve with slow convergence but excellent tracking. The poor tracking in the former case is captured using the PRMSE metric.

High estimation performance should translate to high prediction performance if future inputs are known. In the case of the pneumatic valve, the inputs that describe a complete valve cycle are known, so there is no uncertainty in input prediction for our purposes. Therefore, a low C_w score means that accurate predictions are available sooner, i.e., a *prognostics horizon* is arrived at earlier. A low PRMSE score means that on average, estimates of the wear rate are closer to the true value, so point predictions of EOL and RUL will on average be closer to their true values. Accordingly, a low $\overline{\text{RMAD}}_w$ score translates to more precise predictions.

We summarize prediction performance in the α-λ metric [13], [14]. Here, $\alpha \in [0,1]$ defines bounds as a function of the true RUL, i.e., $(1 \pm \alpha)RUL^*$, and $\lambda \in [0,1]$ denotes the fraction of the time from the first prediction to the true EOL. We use the extended version of the metric, which incorporates a third parameter, $\beta \in [0,1]$, which defines a bound on the fraction of the probability mass of a prediction that falls within the α-bounds [14]. The α-λ metric requires that for given prediction times, a desired portion of the predicted RUL distribution fits within a cone of accuracy defined by α. The α-λ plot for $M=\{x,f,p_t,p_b\}$ is shown in Fig. 6, where α=0.1 and β=0.4 for all λ (all k_P). The distribution at each prediction point is represented using a box plot. The percentage of the probability mass that falls within the α-bounds is shown above each box plot, along with the result of the metric at that point (true or false). As shown in the figure, the desired performance criteria are met at each prediction point except for the 60-cycle prediction point. Note, however, that at this point, the median prediction is within the α-bounds, and the mean is only just outside. With α=0.11, the metric is satisfied at all prediction points. Fig. 7 shows the α-λ plot for $M=\{open,closed\}$. Although the mean and median are typically within the α-bounds, on average only about 32% of the probability mass is contained within the bounds. With α=0.18, the metric is satisfied at all prediction points.

For a particular prediction point k_P, we compute measures of accuracy and prediction. For accuracy, we use the relative accuracy (RA) metric:

$$RA_{k_P} = 1 - \frac{\left| RUL_{k_P}^* - \text{Median}_i(RUL_{k_P}^i) \right|}{RUL_{k_P}^*}$$

Here, we have chosen the median as the measure of central tendency of the distribution. The mean could also
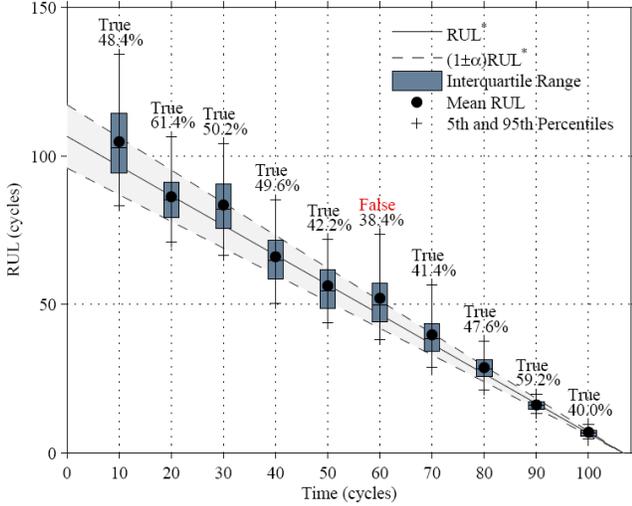
Figure 6 - α-λ metric for spring damage prediction, where α=0.1, β=0.4, and $M=\{x,f,p_t,p_b\}$.

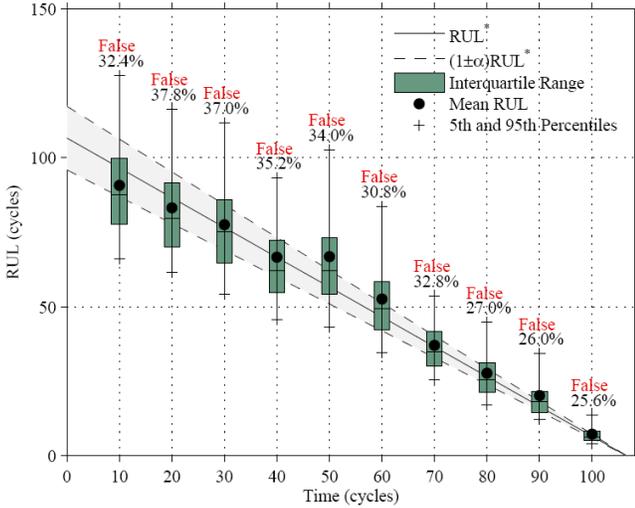**Figure 7 - α-λ metric for spring damage prediction, where α=0.1, β=0.4, and $M=\{open,closed\}$.**

be used, but we have found the median to be more accurate in our experiments. We calculate prediction spread using RMAD, which we denote as RMAD_{RUL}. As a second measure of spread, we compute the fraction of the probability mass that falls within specified α-bounds, denoted as π_α. To summarize these metrics over a particular experiment, we average them over all prediction points. We use $\overline{RA}$ to denote the averaged RA, $\overline{\text{RMAD}}_{RUL}$ to denote averaged RMAD, and $\overline{\pi}_\alpha$ to denote the average fraction of the probability within the α-bounds. In the current example, for $M=\{x,f,p_t,p_b\}$, $\overline{RA}=0.953$, $\overline{\pi}_\alpha=0.487$, $\overline{\text{RMAD}}_{RUL}=10.23\%$, and for $=\{open,closed\}$, $\overline{RA}=0.934$, $\overline{\pi}_\alpha=0.326$, $\overline{\text{RMAD}}_{RUL}=16.16\%$. We also consider worst-case performance over all prediction points. We use the $^-$

superscript to denote a minimum and the $^+$ superscript to denote a maximum. For $M=\{x,f,p_t,p_b\}$, $RA^-=0.927$, $\pi_\alpha^-=0.384$, and $\text{RMAD}_{RUL}^+=14.81\%$. For $M=\{open,closed\}$, $RA^-=0.901$, $\pi_\alpha^-=0.326$, and $\text{RMAD}_{RUL}^+=21.92$. In these cases, a clear connection is observable between performance in estimation and performance in prediction (e.g., for $M=\{x,f,p_t,p_b\}$, PRMSE=5.11% where $\overline{RA}=0.953$, and $\overline{\text{RMAD}}_w=11.00\%$ where $\overline{\text{RMAD}}_{RUL}=10.23\%$. Both sensor sets have fairly good accuracy, but using $M=\{x,f,p_t,p_b\}$ achieves significantly better precision.

We now present a comprehensive set of simulation results for each sensor set for all considered faults. The results are shown in Table 1. We report PRMSE of the wear parameter (in %), $\overline{\text{RMAD}}_w$ (in %), C_w (approximately in seconds), $\overline{RA}$, RA^-, $\overline{\pi}_\alpha$ for α=0.1, π_α^-, $\overline{\text{RMAD}}_{RUL}$ (in %), and RMAD_{RUL}^+ (in %). The results presented are averaged over 10 experiments per scenario.

Overall, the expected results are obtained. The PRMSE is under 10% for all cases, and is around 5% for most. Correspondingly, RA is, in most cases, above 95%. The $\overline{\pi}_\alpha$ score is typically desired to be at least 50%, but here it is lower. For some faults, the prediction variance is large because the random walk variances were not optimally tuned. In general, this cannot be done beforehand since wear parameters are unknown. In an online setting however, correction loops can be used to reduce this variance over time [9].

In general, it is observed that not much difference exists in relative accuracy over the different sensor sets. In fact, in some cases, estimation of the wear parameter is better using only the discrete position sensors rather than the full set of continuous measurements. This means that accurate point predictions can still be achieved even under limited sensing scenarios. However, the prediction spread is significantly larger with limited sensor sets. This is observed best using $\overline{\pi}_\alpha$. For example, for the bottom external leak, there is a 27% decrease from the full set of continuous measurements to the sensor set containing only the discrete position sensors. Convergence of the wear parameter estimate is also affected, although there is no significant difference for the top external leak, which we attribute to the fact that a top external leak of the selected wear rate is easily visible in the pressure measurements, so convergence is quickly forced for all choices of M.

The results also demonstrate the usefulness of different sensors. The flow measurement effectively provides a redundant position measurement, so dropping either x or f from the sensor set has little effect on performance. Some sensors also give more useful information on specific faults, as observed in the cases were M is a singleton. For

Table 1 - Average performance for each fault under different sensor sets

Fault	M	PRMSE	$\overline{\text{RMAD}}_w$	C_w	$\overline{\text{RA}}$	RA^-	$\bar{\pi}_\alpha$	π_α^-	$\overline{\text{RMAD}}_{RUL}$	RMAD_{RUL}^+
Friction Coefficient	$\{x, f, p_t, p_b\}$	6.12	14.95	26.74	0.960	0.892	0.353	0.282	13.79	17.90
$(w_r = 6.0, EOL^* = 100.7)$	$\{x, p_t, p_b\}$	6.68	16.08	24.01	0.945	0.880	0.336	0.290	14.95	19.69
	$\{f, p_t, p_b\}$	5.93	15.47	25.55	0.943	0.908	0.339	0.264	16.85	27.73
	$\{p_t, p_b\}$	6.91	18.72	30.77	0.943	0.808	0.275	0.212	16.46	20.54
	$\{x\}$	6.70	17.24	23.31	0.928	0.834	0.313	0.268	15.47	18.41
	$\{p_t\}$	7.29	20.31	28.07	0.979	0.935	0.286	0.244	21.08	39.21
	$\{p_b\}$	7.24	19.67	24.46	0.960	0.906	0.288	0.254	17.86	23.43
	$\{open, closed\}$	5.56	24.00	30.98	0.939	0.886	0.223	0.198	24.58	35.08
Spring Rate	$\{x, f, p_t, p_b\}$	5.87	10.90	33.29	0.953	0.885	0.475	0.352	10.79	15.59
$(w_r = 0.14, EOL^* = 106.6)$	$\{x, p_t, p_b\}$	5.50	11.77	31.54	0.952	0.880	0.414	0.297	12.08	15.28
	$\{f, p_t, p_b\}$	5.09	11.04	29.04	0.954	0.872	0.444	0.269	11.24	16.35
	$\{p_t, p_b\}$	5.14	13.14	29.10	0.949	0.895	0.400	0.289	12.88	20.93
	$\{x\}$	6.01	12.37	35.44	0.946	0.889	0.403	0.335	12.40	15.96
	$\{p_t\}$	6.11	14.38	41.58	0.925	0.831	0.340	0.239	14.74	21.45
	$\{p_b\}$	6.45	13.84	37.47	0.943	0.878	0.380	0.304	13.48	18.81
	$\{open, closed\}$	4.87	16.37	44.19	0.941	0.843	0.323	0.245	16.13	21.50
Internal Leak	$\{x, f, p_t, p_b\}$	5.98	10.46	25.13	0.948	0.802	0.449	0.281	10.93	16.27
$(w_r = 10^{-11}, EOL^* = 104.6)$	$\{x, p_t, p_b\}$	5.65	10.22	24.09	0.968	0.872	0.481	0.341	10.77	16.45
	$\{f, p_t, p_b\}$	5.17	10.24	33.20	0.951	0.900	0.454	0.377	10.72	15.91
	$\{p_t, p_b\}$	5.36	10.51	26.55	0.944	0.876	0.456	0.363	10.16	14.18
	$\{x\}$	5.93	13.80	20.51	0.949	0.844	0.366	0.286	14.43	19.96
	$\{p_t\}$	6.23	13.41	29.91	0.936	0.860	0.354	0.286	14.45	21.11
	$\{p_b\}$	5.61	10.70	27.34	0.959	0.910	0.455	0.382	11.03	15.81
	$\{open, closed\}$	4.58	17.36	34.60	0.960	0.912	0.309	0.271	16.86	20.53
Top External Leak	$\{x, f, p_t, p_b\}$	5.33	9.27	21.54	0.957	0.796	0.505	0.233	9.32	16.94
$(w_r = 10^{-8}, EOL^* = 134.7)$	$\{x, p_t, p_b\}$	4.93	9.41	19.83	0.961	0.813	0.504	0.247	9.46	17.26
	$\{f, p_t, p_b\}$	5.06	9.38	19.19	0.961	0.796	0.505	0.239	9.41	16.00
	$\{p_t, p_b\}$	4.00	10.04	20.83	0.969	0.862	0.488	0.274	10.12	17.92
	$\{x\}$	6.78	10.99	17.30	0.947	0.772	0.435	0.196	11.06	18.70
	$\{p_t\}$	4.01	10.08	17.94	0.968	0.868	0.486	0.282	10.16	16.33
	$\{p_b\}$	5.31	13.10	18.96	0.956	0.822	0.382	0.191	13.31	23.79
	$\{open, closed\}$	5.14	14.11	18.82	0.959	0.806	0.360	0.217	14.24	27.40
Bottom External Leak	$\{x, f, p_t, p_b\}$	5.68	11.75	25.39	0.954	0.825	0.415	0.257	11.74	17.84
$(w_r = 10^{-9}, EOL^* = 148.0)$	$\{x, p_t, p_b\}$	5.46	11.86	25.02	0.955	0.825	0.414	0.256	11.84	18.27
	$\{f, p_t, p_b\}$	5.68	11.78	27.35	0.953	0.804	0.413	0.236	11.77	18.55
	$\{p_t, p_b\}$	5.15	11.97	25.92	0.959	0.827	0.412	0.260	11.97	17.74
	$\{x\}$	6.19	13.99	25.25	0.949	0.781	0.362	0.245	13.92	46.10
	$\{p_t\}$	5.87	16.22	29.84	0.951	0.808	0.317	0.212	16.17	57.15
	$\{p_b\}$	5.06	11.94	23.48	0.958	0.840	0.413	0.274	11.92	20.14
	$\{open, closed\}$	5.29	17.35	34.11	0.954	0.818	0.303	0.235	17.23	58.96

changes in the friction coefficient and spring rate, $M=\{x\}$ outperforms $\{p_t\}$, $\{p_b\}$, and also $\{p_t, p_b\}$. This is because the position measurement gives more direct information on changes in friction. Similarly, for the cases with leaks, the pressure measurements are more useful. For the internal leak and bottom external leak, p_b is most useful, and for the top external leak, p_t is most useful.

As observed, even by just using the discrete position sensors, accurate point predictions can still be made. In addition to the uncertainty management capabilities of particle filters, this robust performance is largely due to having a good model of the component and its wear mechanisms, which is why we favor model-based approaches. To investigate how the performance changes when both measurements and the model are less reliable, we ran a set of experiments with increased process and measurement noise. The results are presented in Table 2 for the bottom leak fault. We evaluate performance using the key metrics PRMSE, $\overline{\text{RA}}$, $\bar{\pi}_\alpha$ for $\alpha=0.1$, and $\overline{\text{RMAD}}_{RUL}$. Results were averaged over 15 runs for each choice of M. The baseline noise level is denoted by ξ,

Table 2 - Average performance for the bottom leak fault under different noise levels

M	PRMSE			$\overline{RA}$			$\bar{\pi}_\alpha$			$\overline{RMAD}_{RUL}$		
	ξ	10ξ	100ξ	ξ	10ξ	100ξ	ξ	10ξ	100ξ	ξ	10ξ	100ξ
$\{x, f, p_t, p_b\}$	5.68	8.60	12.87	0.954	0.931	0.899	0.415	0.355	0.275	11.74	13.86	18.14
$\{x, p_t, p_b\}$	5.46	7.77	11.27	0.955	0.935	0.907	0.413	0.358	0.274	11.84	13.75	18.36
$\{f, p_t, p_b\}$	5.68	7.94	11.98	0.953	0.935	0.902	0.413	0.357	0.279	11.77	13.90	18.02
$\{p_t, p_b\}$	5.15	7.20	10.97	0.959	0.939	0.909	0.411	0.359	0.271	11.97	13.96	18.60
$\{x\}$	6.19	8.39	12.49	0.949	0.932	0.896	0.362	0.302	0.233	13.92	17.45	22.66
$\{p_t\}$	5.87	9.03	14.63	0.951	0.920	0.874	0.316	0.249	0.190	16.17	21.20	27.68
$\{p_b\}$	5.06	7.79	10.29	0.958	0.935	0.915	0.412	0.349	0.278	11.92	14.26	18.72
$\{open, closed\}$	5.29	7.03	11.69	0.954	0.941	0.904	0.303	0.284	0.241	17.23	19.12	21.76

which corresponds to about 0.1% process noise and 1% measurement noise. We increased the noise variances by factors of 10 (denoted by 10ξ) and 100 (denoted by 100ξ). The 100ξ noise level corresponds to 1% process noise and 10% measurement noise.

As shown in the table, performance degrades as noise is increased. However, even with 100ξ, the average RA has only decreased to about 90% for all choices of M. Significant reductions in prediction spread are also observed. The difference in performance between the full sensor set and using only the discrete position sensors, though, is reduced in the higher noise scenarios. Performance is the best when the bottom pressure sensor is used, as this provides the most direct information for the bottom leak fault.

6. CONCLUSIONS

In this paper, we developed a general model-based prognostics methodology using particle filters, formulated as a joint state-parameter estimation problem. State-parameter estimates are propagated forward in time to obtain EOL and RUL predictions, based on models that capture the progression of damage over time, characterized by a set of unknown parameters. We evaluated the performance under limited sensor sets using prognostics performance metrics [13]. It was shown that, with an accurate model, accurate predictions can still be made even when only discrete position sensors are utilized. The disadvantage is that prediction spread may increase significantly. We also investigated how performance degrades when model uncertainty (i.e., process noise) and sensor noise are increased. Even with the maximum amount of noise considered, an average of 90% accuracy was still achievable, although under much wider prediction spread.

In future work, we will investigate how methods to decrease prediction spread such as correction loops [9] and fixed-lag filters [5] can improve performance under limited sensor sets. In addition, we assumed only single damage mechanisms were active in our experiments. While the approach presented can still be applied to estimation of multiple wear parameters, a decrease in performance is expected. Investigating that case under limited sensor sets is also of interest.

REFERENCES

[1] M. Schwabacher, "A survey of data-driven prognostics," in Proceedings of the AIAA Infotech@ Aerospace Conference, 2005.

[2] M. Roemer, C. Byington, G. Kacprzynski, and G. Vachtsevanos, "An overview of selected prognostic technologies with reference to an integrated PHM architecture," in Proceedings of the First International Forum on Integrated System Health Engineering and Management in Aerospace, 2005.

[3] C. S. Byington, M. Watson, D. Edwards, and P. Stoelting, "A model-based approach to prognostics and health management for flight control actuators," in Proceedings of the 2004 IEEE Aerospace Conference, vol. 6, March 2004, pp. 3551–3562.

[4] B. Saha and K. Goebel, "Modeling Li-ion battery capacity depletion in a particle filtering framework," in Proceedings of the Annual Conference of the Prognostics and Health Management Society 2009, Sept. 2009.

[5] M. Daigle and K. Goebel, "Model-based prognostics with fixed-lag particle filters," in Proceedings of the Annual Conference of the Prognostics and Health Management Society 2009, Sept. 2009.

[6] M. Abbas, A. A. Ferri, M. E. Orchard, and G. J. Vachtsevanos, "An intelligent diagnostic/prognostic framework for automotive electrical systems," in 2007 IEEE Intelligent Vehicles Symposium, 2007, pp. 352–357.

[7] E. Bechhoefer, "A method for generalized prognostics of a component using Paris law," in Proceedings of 64th American Helicopter Society Annual Forum, Apr. 2008.

[8] F. Cadini, E. Zio, and D. Avram, "Monte Carlo-based filtering for fatigue crack growth estimation," Probabilistic Engineering Mechanics, vol. 24, no. 3, pp. 367–373, 2009.

[9] M. Orchard, G. Kacprzynski, K. Goebel, B. Saha, and G. Vachtsevanos, "Advances in uncertainty representation and management for particle filtering applied to prognostics," in Proceedings of International Conference on Prognostics and Health Management, Oct. 2008.

[10] P. Li, R. Goodall, P. Weston, C. Seng Ling, C. Goodman, and C. Roberts, "Estimation of railway vehicle suspension parameters for condition monitoring," Control Engineering Practice, vol. 15, no. 1, pp. 43–55, 2007.

[11] B. Saha, K. Goebel, S. Poll, and J. Christophersen, "Prognostics methods for battery health monitoring using a bayesian framework," IEEE Transactions on Instrumentation and Measurement, vol. 58, no. 2, pp. 291–296, Feb. 2009.

[12] M. E. Orchard, L. Tang, K. Goebel, and G. Vachtsevanos, "A novel RSPF approach to prediction of high-risk, low-probability failure events," in Proceedings of the Annual Conference of the Prognostics and Health Management Society 2009, Sept. 2009.

[13] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, "Metrics for evaluating performance of prognostic techniques," in International Conference on Prognostics and Health Management 2008, Oct 2008.

[14] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "On applying the prognostic performance metrics," in Proceedings of the Annual Conference of the Prognostics and Health Management Society 2009, Sept. 2009.

[15] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," IEEE Transactions on Signal Processing, vol. 50, no. 2, pp. 174–188, 2002.

[16] O. Cappe, S. J. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential Monte Carlo," Proceedings of the IEEE, vol. 95, no. 5, p. 899, 2007.

[17] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," Journal of Computational and Graphical Statistics, vol. 5, no. 1, pp. 1–25, 1996.

[18] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," Statistics and Computing, vol. 10, pp. 197–208, 2000.

[19] R. Perry and D. Green, Perry's chemical engineers' handbook, New York: McGraw-Hill Professional, 2007.

[20] I. M. Hutchings, Tribology: friction and wear of engineering materials, Boca Raton, FL: CRC Press, 1992.

BIOGRAPHY

Matthew Daigle received the B.S. degree in Computer Science and Computer and Systems Engineering from Rensselaer Polytechnic Institute, Troy, NY, in 2004, and the M.S. and Ph.D. degrees in Computer Science from Vanderbilt University, Nashville, TN, in 2006 and 2008, respectively. From September 2004 to May 2008, he was a Graduate Research Assistant with the Institute for Software Integrated Systems and Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN. During the summers of 2006 and 2007, he was an intern with Mission Critical Technologies, Inc., at NASA Ames Research Center. Since June 2008, he has been with the University of California, Santa Cruz, at NASA Ames Research Center as an Associate Research Scientist. His research interests include physics-based modeling, model-based diagnosis and prognosis, simulation, and hybrid systems.

Kai Goebel received the degree of Diplom-Ingenieur from the Technische Universität München, Germany in 1990. He received the M.S. and Ph.D. from the University of California at Berkeley in 1993 and 1996, respectively. Dr. Goebel is a

12

senior scientist at NASA Ames Research Center where he leads the Diagnostics & Prognostics groups in the Intelligent Systems division. In addition, he directs the Prognostics Center of Excellence and he is the Associate Principal Investigator for Prognostics of NASA's Integrated Vehicle Health Management Program. He worked at General Electric's Corporate Research Center in Niskayuna, NY from 1997 to 2006 as a senior research scientist. He has carried out applied research in the areas of artificial intelligence, soft computing, and information fusion. His research interest lies in advancing these techniques for real time monitoring, diagnostics, and prognostics. He holds eleven patents and has published more than 100 papers in the area of systems health management.