An Integrated Model-Based Diagnostic and Prognostic Framework

Indranil Roychoudhury[1] **and Matthew Daigle**[2]

[1] *SGT Inc., NASA Ames Research Center, Moffett Field, CA, 94035, USA*
indranil.roychoudhury@nasa.gov
[2] *University of California, Santa Cruz, NASA Ames Research Center, Moffett Field, CA, 94035, USA*
matthew.j.daigle@nasa.gov

ABSTRACT

Systems health monitoring is essential in guaranteeing the safe, efficient, and correct operation of complex engineered systems. Diagnosis, which consists of detection, isolation and identification of faults; and prognosis, which consists of prediction of the remaining useful life of components, subsystems, or systems; constitute systems health monitoring. This paper presents an integrated model-based diagnostic and prognostic framework, where we make use of a common modeling paradigm to model both the nominal and faulty behavior in all aspects of systems health monitoring. We illustrate our approach using a simulated propellant loading system that includes tanks, valves, and pumps.

1 INTRODUCTION

Systems health monitoring is essential in guaranteeing the safe, efficient, and correct operation of complex engineered systems. The integral tasks of systems health monitoring include both diagnostics and prognostics. Diagnosis involves *detecting* when a fault has occurred, *isolating* the true fault from many possible fault candidates, and *identifying* the true damage to the system. Basically, diagnosis involves determining what *has happened* to the system; while prognosis involves determining what *will happen*. Specifically, prognosis involves *predicting* how much useful life remains in the different components, subsystems, or systems. Based on these predictions, effective actions can be taken to minimize any loss of life or property, optimize maintenance, and extend component life.

A large body of research exists for both diagnostics and prognostics. However, many diagnosis approaches stop at the fault isolation step, and seldom perform fault identification; and most prognostic approaches assume some diagnosis has been performed and focus on prognosis of a single failure mode. This paper presents an integrated model-based framework for diagnostics and prognostics of complex systems, in which we make use of a common modeling framework for modeling both the nominal and faulty system behavior used for both diagnostics and prognostics. We assume only single faults in this paper.

In our approach, we start with modeling the nominal system, as well as how different faults manifest in the system behavior and progress over time. An observer built with the nominal model is used to generate estimates of nominal system behavior, and when the deviation of observed measurements from the nominal estimates is statistically significant, a fault is detected. Fault isolation involves comparing the observed measurement deviations to predictions of how these measurements would deviate for different possible faults, and removing from consideration fault candidates that are inconsistent with the observed deviations. Fault identification involves tracking the observed system measurements using multiple observers, each built with a hypothesized fault model integrated with the nominal model, and performing joint state-parameter estimation (Roychoudhury, 2009). The prognosis module predicts the remaining useful life of a component, subsystem, or system, using, for each hypothesized fault, a predictor based on the fault progression model integrated with the nominal model (Daigle and Goebel, 2011). We perform a number of experiments on a simulated propellant loading system to demonstrate and evaluate our approach.

In this paper, Section 2 provides the problem formulation for our diagnostic and prognostic framework; Section 3 describes the architecture and its different components; Section 4 presents the case study and experimental results; and Section 5 concludes the paper.

2 PROBLEM FORMULATION

We define a system model for representing system behavior under nominal operation as follows:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{v}(t))\\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{n}(t)),\end{aligned}$$

where $t \in \mathbb{R}$ denotes continuous time, $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ is the state vector, $\boldsymbol{\theta}(t) \in \mathbb{R}^{n_\theta}$ is the parameter vector, $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ is the input vector, $\mathbf{v}(t) \in \mathbb{R}^{n_v}$ is the process noise vector, $\mathbf{f}$ is the state equation, $\mathbf{y}(t) \in \mathbb{R}^{n_y}$

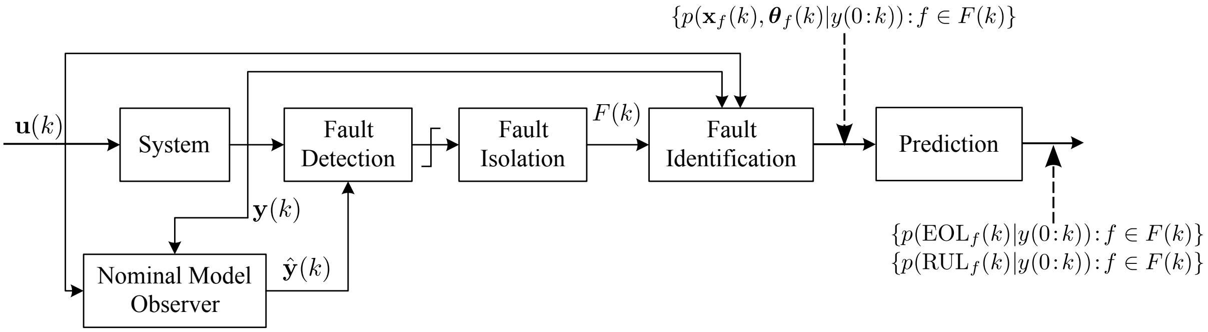
$\{p(\mathbf{x}_f(k), \boldsymbol{\theta}_f(k)|y(0\!:\!k)) : f \in F(k)\}$
$\mathbf{u}(k)$
System
Fault Detection
Fault Isolation
$F(k)$
Fault Identification
Prediction
$\mathbf{y}(k)$
Nominal Model Observer
$\hat{\mathbf{y}}(k)$
$\{p(\mathrm{EOL}_f(k)|y(0\!:\!k)) : f \in F(k)\}$
$\{p(\mathrm{RUL}_f(k)|y(0\!:\!k)) : f \in F(k)\}$

Figure 1: The integrated diagnostic and prognostic architecture.

is the output vector, $\mathbf{n}(t) \in \mathbb{R}^{n_n}$ is the measurement noise vector, and $\mathbf{h}$ is the output equation. The parameters $\boldsymbol{\theta}(t)$ evolve in an unknown way.

Any change in the above nominal system model represents a fault. In this work, we restrict faults solely to changes in system parameters, $\boldsymbol{\theta}(t)$. Under the single fault assumption, only one parameter can deviate from nominal. Hence, we denote a fault, $f \in F$, as a tuple, (θ, g_f), where, $\theta \in \boldsymbol{\theta}$ is the faulty parameter, and g_f denotes the *fault progression function*, according to which, fault f is manifested in parameter θ, i.e.,

$$\dot{\theta}(t) = g_f(t, \mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t), \mathbf{m}_f(t)),$$

where $\mathbf{x}_f(t) = [\mathbf{x}(t), \ \theta(t)]^T$, $\boldsymbol{\theta}_f(t) = [\boldsymbol{\theta}(t) \backslash \{\theta(t)\}, \boldsymbol{\phi}_f(t)]^T$, $\boldsymbol{\phi}_f(t) \in \mathbb{R}^{n_{\phi_f}}$ is a vector of fault progression model parameters, and $\mathbf{m}_f(t) \in \mathbb{R}^{n_{m_f}}$ is a process noise vector associated with the fault progression model.

The single fault assumption also implies that the faulty system model for fault $f = (\theta, g_f)$ involves integrating a single fault progression model into the nominal system model:

$$\begin{aligned} \dot{\mathbf{x}_f}(t) &= \mathbf{f}_f(t, \mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t), \mathbf{v}(t)) \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{n}(t)), \end{aligned}$$

where,

$$\mathbf{f}_f(\cdot) = \begin{bmatrix} \mathbf{f}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{v}(t)) \\ g_f(t, \mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t), \mathbf{m}(t)) \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\theta}(t) \end{bmatrix}.$$

Since any of the several parameters in a system model can be faulty, the goal of diagnosis is to:

1. Detect a change in some $\theta \in \boldsymbol{\theta}$;
2. Isolate, under the single fault assumption, the true $f \in F$, i.e., both the parameter θ that has changed, and its fault progression model g_f; and
3. Identify the extent of damage by computing $p(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t)|\mathbf{y}(0\!:\!t))$, where $\mathbf{y}(0\!:\!t)$ denotes all measurements observed up to time, t.

The goal of prognosis is to predict for a given fault, f, using, $p(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t)|\mathbf{y}(0\!:\!t_P))$, a probability distribution of end of life (EOL), i.e., $p(\mathrm{EOL}_f(t_P)|\mathbf{y}(0\!:\!t_P))$, and/or remaining useful life (RUL), i.e., $p(\mathrm{RUL}_f(t_P)|\mathbf{y}(0\!:\!t_P))$ at a given time point t_P (Daigle and Goebel, 2011). We predict the probability distribution, rather than a single EOL and/or RUL value, since there is inherent uncertainty in the estimation of state, and the future inputs. A set of constraints define the acceptable behavior of a system. The system has failed when one or more of the constraints are no longer met. We define a threshold function, T_{EOL_f}, where $T_{\mathrm{EOL}_f}(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t)) = 1$ if these constraints are valid, and $T_{\mathrm{EOL}_f}(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t)) = 0$ otherwise.

So, EOL_f may be defined as $\mathrm{EOL}_f(t_P) \triangleq \inf\{t \in \mathbb{R} : t \geq t_P \text{ and } T_{\mathrm{EOL}_f}(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t)) = 1\}$. i.e., EOL is the earliest time point at which the threshold is reached. Given $\mathrm{EOL}_f(t_P)$, RUL may then be defined with $\mathrm{RUL}_f(t_P) \triangleq \mathrm{EOL}_f(t_P) - t_P$.

3 DIAGNOSIS AND PROGNOSIS APPROACH

Fig. 1 illustrates the architecture of our combined diagnostic and prognostic scheme. At each discrete time step, k, the system takes as inputs $\mathbf{u}(k)$, and outputs measurements $\mathbf{y}(k)$. The nominal observer also takes as inputs $\mathbf{u}(k)$, and generates estimates of nominal measurements, $\hat{\mathbf{y}}(k)$. The fault detector then takes in the observed and estimated measurements, $\mathbf{y}(k)$ and $\hat{\mathbf{y}}(k)$, and detects when a fault has occurred based on the residual, $\mathbf{r}(k) = \mathbf{y}(k) - \hat{\mathbf{y}}(k)$. Once a fault is detected, fault isolation is initiated. The fault isolation block takes as inputs $\mathbf{r}(k)$. These measurement residuals are used along with predictions of how each measurement is expected to deviate from nominal for each possible fault in the system to generate a set of fault candidates $F(k)$ at time k that explain the observed deviations in measurements till time k. The fault identification module, for each fault, $f \in F(k)$, estimates $p(\mathbf{x}_f(k), \boldsymbol{\theta}_f(k)|\mathbf{y}(0\!:\!k))$. Finally, the prediction module takes as input $p(\mathbf{x}_f(k), \boldsymbol{\theta}_f(k)|\mathbf{y}(0\!:\!k))$ to make predictions of EOL, i.e., $p(\mathrm{EOL}_f(k)|y(0\!:\!k))$, and/or RUL, i.e., $p(\mathrm{RUL}_f(k)|y(0\!:\!k))$.

The remainder of this section describes the details of the different modules of the integrated diagnosis and prognosis architecture.

3.1 Nominal Observer

The nominal observer takes as inputs the system inputs, $\mathbf{u}(k)$, and measurements, $\mathbf{y}(0\!:\!k)$, and the initial state of the system, and uses the state transition function, $\mathbf{f}(\cdot)$, and observation function, $\mathbf{h}(\cdot)$, to estimate distributions of states, $\mathbf{x}(k)$, and parameters, $\boldsymbol{\theta}(k)$, i.e., $p(\mathbf{x}(k), \boldsymbol{\theta}(k)|\mathbf{y}(0\!:\!k))$.

Any appropriate filtering scheme, e.g., Kalman filter, extended Kalman filter, unscented Kalman filter, particle filter (Arulampalam *et al.*, 2002), among others, can be adopted as the nominal observer.

3.2 Fault Detection

A fault is detected when a residual, $r(k) \in \mathbf{r}(k)$, i.e., the difference between the observed (faulty) and estimated (nominal) values of a measurement, is determined to be statistically significant (Daigle *et al.*, 2010). In our work, we use a Z-test coupled with a sliding window technique to determine this statistical significance (Daigle *et al.*, 2010).

3.3 Fault Isolation

Once a fault is detected, at each subsequent time step, every measurement residual is qualitatively abstracted into a tuple of qualitative symbols, (σ_1, σ_2), where $\sigma_1 \in \{0, +, -\}$ represents the qualitative magnitude change, and $\sigma_2 \in \{0, +, -\}$ represents the qualitative slope change. The symbols, 0, +, or −, denote whether the magnitude or slope of this measurement is at, above, or below nominal, respectively. The symbols are generated using a sliding window technique as described in detail in (Daigle *et al.*, 2010).

Based on the first observed statistically significant measurement deviation, we generate a set of possible fault candidates. Then, for each fault candidate, we systematically determine a fault signature for each measurement (Mosterman and Biswas, 1999). A fault signature of a fault for a measurement is a prediction of how the measurement will deviate from nominal due to the fault. Fault signatures are also of the form (s_1, s_2), where $s_1 \in \{0, +, -\}$ and $s_2 \in \{0, +, -\}$ capture qualitatively the direction of change to be expected in the magnitude and slope of each measurement from nominal if the fault occurs.

In addition to fault signatures, we also make use of relative measurement orderings (Daigle *et al.*, 2007). Measurement orderings encode information about the temporal order in which fault effects will manifest in different measurements. They can be determined by analyzing the transfer functions from faults to measurements (Daigle *et al.*, 2007). If fault f manifests in measurement m_i before measurement m_j, then a relative measurement ordering can be defined between m_i and m_j for fault f, and is denoted by $m_i \prec_f m_j$.

Given the set of fault candidates, as measurements deviate from nominal, the observed measurement deviations (captured symbolically) are checked for consistency with predicted fault signatures and measurement orderings. Any fault candidate whose predictions are inconsistent is removed from consideration. As more and more measurement deviations are observed, the candidate set will reduce, ideally resulting in a singleton.

However, in some cases, the qualitative fault signatures alone are not sufficient in distinguishing all faults, or fault effects may take too long to manifest, and quantitative analysis is needed to correctly diagnose the true fault. The advantage of using qualitative fault isolation is that it reduces the fault candidates very quickly, thereby improving the scalability of the overall diagnosis task. Hence, the more diagnosable the system is, the smaller is the number possible fault candidates remaining after fault isolation is performed, and fewer will be the faults that will have to be isolated through relatively expensive quantitative methods.

3.4 Fault Identification

We initiate quantitative fault identification after qualitative fault signature-based isolation is executed for p time steps or till the number of fault candidates reduces to less than σ, whichever is achieved first. The design parameters p and σ are chosen based on the design requirements of the integrated diagnostic and prognostic system.

Once fault identification is invoked, under the single fault assumption, for each remaining fault candidate, f, we instantiate an observer using its faulty system model, $\mathbf{f}_f(\cdot)$ and $\mathbf{h}(\cdot)$, generated, as described in Section 2, by extending the nominal system model with the fault progression model. Then each fault observer tracks the observed system measurements independently, and generates estimates of $\hat{\mathbf{y}}(k)$ and $p(\mathbf{x}_f(k), \boldsymbol{\theta}_f(k) | \mathbf{y}(k_d - \Delta k^{max} : k))$, Δk^{max} is usually assumed to be larger than the time difference between the time of fault occurrence, k_f, and the time of fault detection, k_d. Each fault observer is initialized to estimated values of $\mathbf{x}$ and $\boldsymbol{\theta}$ obtained from the nominal observer at time $k_d - \Delta k^{max}$, and $\boldsymbol{\phi}_f$ is initialized to a zero vector. If multiple fault candidates remain when fault identification is invoked, for each fault observer, a Z-test is used to determine if the deviation of a measurement estimated by the observer from the corresponding actual observation is statistically significant. Since we are considering only single faults, the expectation is that eventually, the estimates of only the correct fault observer will converge to the observed measurements, while those of all others will deviate from the observed measurements. Thus fault identification also helps in fault isolation. Practically, even the true fault model will take some time before tracking the measurements correctly, since initially, the fault parameter values are most likely to be incorrect. We assume that the true fault observer will converge to the observed measurements within s_d time steps of its invocation. Thus, the Z-tests are monitored only after s_d time steps are over (Roychoudhury, 2009).

3.5 Prediction

The prediction module is invoked at time k_P to predict the EOL and/or RUL of the component for each hypothesized fault, f. Specifically, using the current joint state-parameter estimate, $p(\mathbf{x}_f(k_P), \boldsymbol{\theta}_f(k_P) | \mathbf{y}(0\!:\!k_P))$, which represents the most up-to-date knowledge of the system at time k_P, the goal is to compute $p(\mathrm{EOL}_f(k_P) | \mathbf{y}(0\!:\!k_P))$ and $p(\mathrm{RUL}_f(k_P) | \mathbf{y}(0\!:\!k_P)$. We assume the state-parameter distribution is represented as a discrete set of weighted samples, i.e.,

$$p(\mathbf{x}_f(k_P), \boldsymbol{\theta}_f(k_P) | \mathbf{y}(0\!:\!k_P)) \approx$$
$$\sum_{i=1}^{N} w^i(k_P) \delta_{(\mathbf{x}^i_f(k_P), \boldsymbol{\theta}^i_f(k_P))}(d\mathbf{x}_f(k_P) d\boldsymbol{\theta}_f(k_P)),$$

where i denotes the index of a single sample, w^i is the weight of this sample, and δ represents the Dirac delta function located at $(\mathbf{x}^i_f(k_P), \boldsymbol{\theta}^i_f(k_P))$.

Algorithm 1 EOL Prediction

```
**Inputs:** $\{(\mathbf{x}_f^i(k_P), \boldsymbol{\theta}_f^i(k_P)), w^i(k_P)\}_{i=1}^N$
**Outputs:** $\{EOL_f^i(k_P), w^i(k_P)\}_{i=1}^N$
**for** $i = 1$ **to** $N$ **do**
    $k \leftarrow t_P$
    $\mathbf{x}_f^i(k) \leftarrow \mathbf{x}_f^i(k_P)$
    $\boldsymbol{\theta}_f^i(k) \leftarrow \boldsymbol{\theta}_f^i(k_P)$
    **while** $T_{\text{EOL}_f}(\mathbf{x}_f^i(k), \boldsymbol{\theta}_f^i(k)) = 0$ **do**
        Predict $\hat{\mathbf{u}}(k)$
        $\boldsymbol{\theta}_f^i(k+1) \sim p(\boldsymbol{\theta}_f(k+1)|\boldsymbol{\theta}_f^i(k))$
        $\mathbf{x}_f^i(k+1) \sim p(\mathbf{x}_f(k+1)|\mathbf{x}_f^i(k), \boldsymbol{\theta}_f^i(k), \hat{\mathbf{u}}(k))$
        $k \leftarrow k+1$
        $\mathbf{x}_f^i(k) \leftarrow \mathbf{x}_f^i(k+1)$
        $\boldsymbol{\theta}_f^i(k) \leftarrow \boldsymbol{\theta}_f^i(k+1)$
    $\text{EOL}_f^i(k_P) \leftarrow k$
```

Similarly, we can approximate the EOL as

$$p(\text{EOL}_f(k_P)|\mathbf{y}(0\!:\!k_P) \approx \sum_{i=1}^{N} w^i(k_P)\delta_{\text{EOL}_f^i(k_P)}(d\text{EOL}_f(k_P)).$$

The general approach to solving the prediction problem is through simulation. Each sample is simulated forward to EOL to obtain the complete EOL distribution. The pseudocode for the prediction procedure is given as Algorithm 1 (Daigle and Goebel, 2011). Each sample i in the state-parameter distribution is propagated forward until $T_{\text{EOL}_f}(\mathbf{x}_f^i(k), \boldsymbol{\theta}_f^i(k))$ evaluates to 1, at which point EOL has been reached for this particle, and the EOL prediction is weighted by the weight of the sample at k_P.

Note that we need to hypothesize future inputs of the system, $\hat{\mathbf{u}}(k)$, for prediction, since fault progression is dependent on the operational conditions of the system. The choice of expected future inputs depends on the knowledge of expected operational settings.

4 CASE STUDY

We apply the approach to a simulation of a propellant loading system. The system schematic is shown in Fig. 2 and is based on a subset of the system presented in (Goodrich *et al.*, 2009). Liquid is drained from a storage tank through a transfer line via a pump, into a vehicle tank. In normal operation, both valves V_1 and V_2 on the transfer line are fully open, and the valve V_3 on the recirculation line is fully closed.

In our case study, we assume there is a loading schedule that defines what valves to open and what RPM to run the pump at. We are interested in predicting how many fueling operations a component will last. To this end, this schedule is maintained throughout the experiment, as back-to-back loading scenarios are simulated. Hence, in this work, we assume future inputs are known. However, in general, development of efficient methods for handling future input uncertainty is still an open problem in prognostics, with some examples being, assuming a single fixed trajectory; or assuming several fixed trajectories, and weighting them by probability, predicting each trajectory, and finally weighting these predictions.

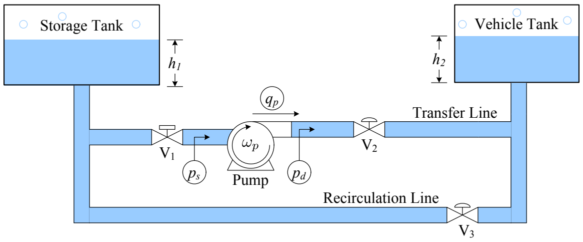

Figure 2: Fueling system schematic.

System measurements include the tank heights, h_1 and h_2, the suction and discharge pressures of the pump, p_s and p_d, the rotational velocity of the pump, ω_p, the discharge flow of the pump, q_p, and the thrust bearing, radial bearing, and oil temperatures of the pump, T_t, T_r, and T_o, respectively (the location of temperature sensors are not shown in Fig. 2).

In the remainder of this section, we first describe the system model. We then provide an example scenario to demonstrate the approach, followed by a summary of diagnosis and prognosis results.

4.1 Nominal System Modeling

The storage and vehicle tank masses are described by

$$\dot{m}_1(t) = q_{V3} - q_{V1} - q_{l1}$$
$$\dot{m}_2(t) = q_{V2} - q_{V3} - q_{l2},$$

where the flows q_{Vi} through valve V_i are defined as

$$q_{V1} = u_{V1}A_{V1}\sqrt{|p_1 - p_s|}\text{sign}(p_1 - p_s)$$
$$q_{V2} = u_{V2}A_{V2}\sqrt{|p_d - p_2|}\text{sign}(p_d - p_2)$$
$$q_{V3} = u_{V3}A_{V3}\sqrt{|p_2 - p_1|}\text{sign}(p_2 - p_1)$$

such that $u_{Vi} \in [0, 1]$ denotes the commanded position of valve V_i with 0 denoting the valve is fully closed, and 1 denoting the valve is fully open; and A_C denotes the product of the cross-sectional area of component C and its flow coefficient, with leakage flows:

$$q_{l1} = A_{l1}\sqrt{|p_1 - p_{atm}|}\text{sign}(p_1 - p_{atm})$$
$$q_{l2} = A_{l2}\sqrt{|p_2 - p_{atm}|}\text{sign}(p_2 - p_{atm}).$$

Leakage areas A_{l1} and A_{l2} are nominally 0.

The tank pressures are given by

$$p_1 = p_{atm} + \rho g h_1$$
$$p_2 = p_{atm} + \rho g h_2,$$

with $h_j = m_j/(\rho A_j)$, where ρ is the liquid density and A_j is the tank cross-sectional area (the tanks are assumed to have a uniform cross-sectional area). The suction and discharge pressures are given by

$$\dot{p}_s = 1/C_s(q_{V1} - q_p)$$
$$\dot{p}_d = 1/C_d(q_p - q_{p2}),$$

where C_s and C_d are pipe capacitances, and q_p is the pump flow.

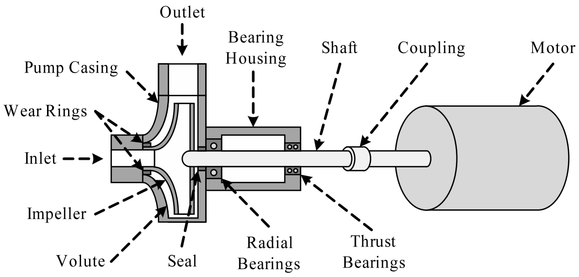

Figure 3: Centrifugal pump.

The centrifugal pump takes in fluid through its inlet, and the rotation of its impellar forces the fluid through the outlet. Fig. 3 presents the schematic of a centrifugal pump.

The rotational velocity of the pump is described using a torque balance,

$$\dot{\omega_p} = \frac{1}{J}\left(\tau_e(t) - r\omega_p(t) - \tau_L(t)\right),$$

where J is the lumped motor/pump inertia, τ_e is the electromagnetic torque provided by the motor, r is the lumped friction parameter, and τ_L is the load torque. A torque is produced on the rotor only when there is a difference (i.e., slip) between the synchronous speed of the supply voltage, ω_s and the mechanical rotation, ω_p, where slip s is defined as

$$s = \frac{\omega_s - \omega_p}{\omega_s}.$$

The expression for the torque τ_e for an alternating-current induction motor is (Lyshevski, 1999):

$$\tau_e = \frac{npR_2}{s\omega_s}\frac{V_{rms}^2}{(R_1 + R_2/s)^2 + (\omega_s L_1 + \omega_s L_2)^2},$$

where R_1 is the stator resistance, L_1 is the stator inductance, R_2 is the rotor resistance, L_2 is the rotor inductance, n is the number of phases, and p is the number of pole pairs. Rotor speed is controlled by changing the input frequency ω_s.

The load torque τ_L is a polynomial function of the flow rate through the pump and the impeller rotational velocity (Kallesøe, 2005):

$$\tau_L = a_0\omega_p^2 + a_1\omega_p q_p - a_2 q_p^2,$$

where q_p is the pump flow, and a_0, a_1, and a_2 are coefficients derived from the pump geometry.

The rotation of the impeller creates a pressure difference from the inlet to the outlet of the pump, driving the pump flow, q_p. The resulting pump pressure is

$$p_p = b_0\omega_p^2 + b_1\omega_p q_p - b_2 q_p^2,$$

where b_0, b_1, and b_2 are coefficients derived from the pump geometry. The parameter b_0 is proportional to impeller area A_i. Flow through the impeller, q_i, is computed using the pressure differences:

$$q_i = c\sqrt{|p_s + p_p - p_d|}sign(p_s + p_p - p_d),$$

where c is a flow coefficient, p_s is the suction pressure, and p_d is the discharge pressure. The small (normal) leakage flow from the discharge end to the suction end due to the clearance between the wear rings and the impeller is described by

$$q_l = c_l\sqrt{|p_d - p_s|}sign(p_d - p_s),$$

where c_l is a flow coefficient. The discharge flow, q_p, is then

$$q_p = q_i - q_l.$$

Pump temperatures are often monitored as indicators of pump condition. The oil heats up due to the radial and thrust bearings and cools to the environment:

$$\dot{T_o} = \frac{1}{J_o}(H_{o,1}(T_t - T_o) + H_{o,2}(T_r - T_o) - H_{o,3}(T_o - T_a)),$$

where J_o is the thermal inertia of the oil, and the $H_{o,i}$ terms are heat transfer coefficients. The thrust bearings heat due to the friction between the pump shaft and the bearings, and cool to the oil and the environment:

$$\dot{T_t} = \frac{1}{J_t}(r_t\omega^2 - H_{t,1}(T_t - T_o) - H_{t,2}(T_t - T_a)),$$

where J_t is the thermal inertia of the thrust bearings, r_t is the friction coefficient for the thrust bearings, and the $H_{t,i}$ terms are heat transfer coefficients. The radial bearings behave similarly:

$$\dot{T_r} = \frac{1}{J_r}(r_r\omega^2 - H_{r,1}(T_r - T_o) - H_{r,2}(T_r - T_a))$$

where J_r is the thermal inertia of the radial bearings, r_r is the friction coefficient for the radial bearings, and the $H_{r,i}$ terms are heat transfer coefficients. Please refer to (Daigle and Goebel, 2011) for additional details on pump modeling.

4.2 Faulty System Modeling

We consider the eight faults shown in Table 1. Either tank can have a leak fault, represented as an abrupt increase in parameter A_{l1} or A_{l2}. For tank i, the abrupt increase in A_{li} is characterized by the fault progression function,

$$\dot{A}_{li} = \begin{cases}\delta(t)\Delta A_{li}, & t = t_f\\ 0, & \text{otherwise}\end{cases}$$

where δ is a Dirac delta function, t_f is the time of fault occurrence, and ΔA_{li} is the fault parameter.

Valves V_1 and V_2 are nominally open and valve V_3 is nominally closed. Hence, stuck faults in these three valves are denoted by x_1^-, x_2^-, and x_3^+ where each Δx_i denotes the difference in the value at which valve V_i gets abruptly stuck at and its nominal value. Therefore, the fault progression function for valve V_i is

$$\dot{x}_i = \begin{cases}\delta(t)\Delta x_i, & t = t_f\\ 0, & \text{otherwise.}\end{cases}$$

For these abrupt faults, the component is assumed to have reached its EOL, i.e., $T_{\text{EOL}_f} = 1$, as soon as the fault occurs, i.e., as soon as a leak is present in a tank,

Table 1: Faults of Interest

Fault Name	Description	θ	g_f	ϕ_f
A_{l1}^+	Leak in storage tank	A_{l1}	$\dot{A}_{l1} = \begin{cases} \delta(t)\Delta A_{l1}, & t = t_f \\ 0, & \text{otherwise} \end{cases}$	ΔA_{l1}
A_{l2}^+	Leak in vehicle tank	A_{l2}	$\dot{A}_{l2} = \begin{cases} \delta(t)\Delta A_{l2}, & t = t_f \\ 0, & \text{otherwise} \end{cases}$	ΔA_{l2}
x_1^-	V$_1$ stuck at x_1	x_1	$\dot{x}_1 = \begin{cases} \delta(t)\Delta x_1, & t = t_f \\ 0, & \text{otherwise} \end{cases}$	Δx_1
x_2^-	V$_2$ stuck at x_2	x_2	$\dot{x}_2 = \begin{cases} \delta(t)\Delta x_2, & t = t_f \\ 0, & \text{otherwise} \end{cases}$	Δx_2
x_3^+	V$_3$ stuck at x_3	x_3	$\dot{x}_3 = \begin{cases} \delta(t)\Delta x_3, & t = t_f \\ 0, & \text{otherwise} \end{cases}$	Δx_3
A_i^-	Impellar wear	A_i	$\dot{A}_i(t) = -w_{A_i} q_i(t)^2$	w_{A_i}
r_t^+	Thrust bearing wear	r_t	$\dot{r}_t(t) = w_t r_t \omega^2$	w_t
r_r^+	Radial bearing wear	r_r	$\dot{r}_r(t) = w_r r_r \omega^2$	w_r

or a valve becomes stuck. As a result, RUL predictions associated with these components are trivially 0 whenever they are diagnosed.

Faults in the pump are not abrupt, but incipient, i.e., they progress slowly, and include impeller wear, A_i^-, represented as a progressive decrease in impeller area A_i (Biswas and Mahadevan, 2007; Daigle and Goebel, 2011); and bearing wear faults, r_t^+ and r_r^+, represented as progressive changes in the thrust bearing friction coefficient, r_t, or the radial bearing friction coefficient, r_r, respectively (Daigle and Goebel, 2011).

Since the impeller area is proportional to b_0, a decrease in it causes a decrease in the pump pressure, and hence, the pump efficiency. The equation to describe how the impeller area decreases over time (Daigle and Goebel, 2011) based on the erosive wear equation (Hutchings, 1992) is as follows:

$$\dot{A}_i(t) = -w_{A_i} q_i(t)^2.$$

Bearing wear is based on sliding and rolling wear equations (Hutchings, 1992; Daigle and Goebel, 2011):

$$\dot{r}_t(t) = w_t r_t \omega^2$$
$$\dot{r}_r(t) = w_r r_r \omega^2,$$

where w_t and w_r are the wear coefficients.

The pump is still functional, i.e., it is still delivering fluid, in the presence of the three wear faults. Hence, its EOL is defined by the effective impeller area deceasing to a certain level $A_i^\downarrow$, and by its temperatures exceeding given thresholds at which irreversible damage occurs, $T_t^\uparrow$, $T_r^\uparrow$, or $T_o^\uparrow$, where abnormal temperature increases are related to increases in bearing friction. So, for a pump fault $f \in F$, $T_{\text{EOL}_f} = 1$ if $A_i(t) \leq A_i^\downarrow$, $T_t(t) \geq T_t^\uparrow$, $T_r(t) \geq T_r^\uparrow$, or $T_o(t) \geq T_o^\uparrow$.

4.3 Demonstration of Approach

We now present a detailed integrated diagnosis and prognosis scenario to illustrate the approach. The fault signatures and some selected measurement orderings are given in Table 2. We assume all random variables to be Gaussian.

For our case study, we adopt the particle filter as our filtering scheme. Particle filtering is the most general estimation scheme as it can be applied to nonlinear systems with arbitrary probability distributions for process and measurement noise that can be nonlinearly coupled with the states. Particle filtering is a sequential Monte Carlo sampling method for Bayesian filtering and approximates the belief state of a system using a weighted set of samples, or particles. Each particle consists of an instantiation of values of the state vector, and describes a possible system state. As observations are obtained, each particle is moved stochastically to a new state using the nominal state transition function, and the weight of each particle is readjusted to reflect the likelihood of that observation given the particle's new state.

In this scenario, impeller wear begins at $t = 0$ s with wear rate $w_A = 3 \times 10^{-3}$. A fault is detected at 934 s, via a decrease in the pump flow q_p. The initial candidate list is reduced to $\{A_i^-, x_1^-, x_2^-\}$ based on the signatures and orderings. At 2729 s, a decrease in h_2 is detected, eliminating x_1^- since that fault would have caused a deviation in p_s before h_2. At 3117 s, an increase in h_1 is detected, eliminating x_2^- since that fault would have produced a change in p_d before h_1. Thus the true fault is isolated.

Fault identification was initiated once the number of fault candidates was reduced to three or less (i.e., $\sigma = 3$) by the qualitative isolator, or if the qualitative isolator has executed for $p = 3000$ s. For our particular problem, we found $N = 50$ particles sufficient for accurate tracking, and used $\Delta k^{max} = 0$ for each observer used for fault identification. For the impeller wear fault, the wear rate w_A estimate averaged to $w_A = 3.13 \times 10^{-3}$ with small output error. Fig. 4 shows the estimated wear parameter estimate for impeller wear. Because the fault progression is so slow, by the end of the first fueling (at $10,000$ s) the estimate is still converging. In further fuelings the estimate has converged with a small spread and remains fairly steady, due to the use of the variance control algorithm presented in (Daigle and Goebel, 2011) that dynamically modifies the random walk variance of the prediction algorithm to maintain a user-specified relative spread of the unknown fault parameters. The corresponding RUL predictions, made at the halfway point and the end of each fueling are shown in Fig. 5. By the third prediction point, the algorithm has converged and predictions remain within the desired accuracy window of 10%. The predictions were made assuming known future system inputs, so the uncertainty in the predictions is due solely to that resulting from the identification stage.

4.4 Simulation Results

Table 3 summarizes the results of several simulation experiments. The columns of the table represent the true fault; true injected value of the fault parameter ϕ_f; k_f, the time of fault occurrence in seconds from the start of experiment; Δk_d, the time in seconds to detect the fault; Δk_i, the time in seconds for qualitative isolation to reduce the candidate set as much as possible;

Table 2: Fault signatures and selected measurement orderings.

Faults	h_1	h_2	p_s	p_d	ω_p	q_p	T_t	T_r	T_o	Measurement Orderings
A_{l1}^+	0−	0−	0−	0+	0−	0−	0−	0−	0−	$h_1 \prec q_p, h_1 \prec h_2, h_1 \prec T_t, h_1 \prec T_r$
A_{l2}^+	0−	0−	0−	0−	0+	0+	0+	0+	0+	$h_2 \prec q_p, h_2 \prec h_1, h_2 \prec T_t, h_2 \prec T_r$
x_1^-	0+	0−	0−	0−	0−	0−	0−	0−	0−	$p_s \prec h_2, p_s \prec T_t, p_s \prec T_r$
x_2^-	0+	0−	0+	0+	0−	0−	0−	0−	0−	$p_d \prec h_1, p_d \prec T_t, p_d \prec T_r$
x_3^+	0−	0+	0−	0+	0−	0−	0−	0−	0−	$h_1 \prec q_p, h_2 \prec q_p, h_1 \prec T_t, h_1 \prec T_r$
A_i^-	0+	0−	0+	0−	0−	0−	0−	0−	0−	$q_p \prec h_1, q_p \prec h_2, q_p \prec T_t, q_p \prec T_r$
r_t^+	0+	0−	0+	0−	0−	0−	0+	0+	0+	$T_t \prec T_o, T_o \prec T_r, T_t \prec Q, T_t \prec h_1, T_t \prec h_2$
r_r^+	0+	0−	0+	0−	0−	0−	0+	0+	0+	$T_r \prec T_o, T_o \prec T_t, T_r \prec Q, T_r \prec h_1, T_r \prec h_2$

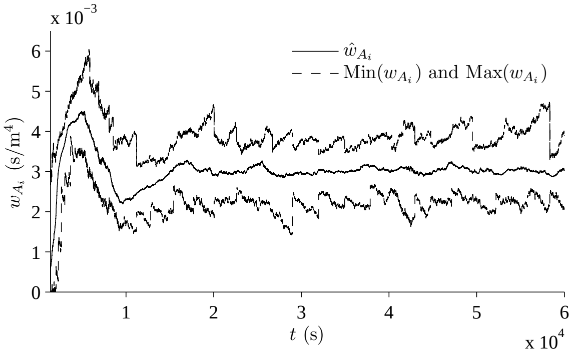

Figure 4: Estimated w_{A_i} values.

the set of fault candidates after qualitative fault isolation; the true fault parameter identified after qualitative fault isolation and quantitative identification; $\overline{\text{RA}}$, the relative accuracy (RA) averaged over every prediction point, where RA is defined as

$$\text{RA}_{k_P} = 100 \left(1 - \frac{|\text{RUL}^*_{k_P} - \widehat{\text{RUL}}_{k_P}|}{\widehat{\text{RUL}}_{k_P}} \right),$$

such that $\text{RUL}^*_{k_P}$ is the true RUL at time k_P, and $\widehat{\text{RUL}}_{k_P}$ is the mean of the prediction (Saxena *et al.*, 2010); and the maximum number of fault identifiers running concurrently. For the abrupt faults, EOL is reached as soon as the fault is detected, and hence, $\overline{\text{RA}}$ is not applicable. For the pump wear faults, however, the EOL is reached when certain thresholds are reached, and so EOL/RUL predictions are performed every half loading cycle, or 5000 s, and $\overline{\text{RA}}$ is computed.

In most cases, the faults were isolated fairly quickly after detection, mainly due to the diagnostic power of the measurement orderings. E.g., if T_t deviates first then the only consistent fault is r_t^+. Isolation times were slower when more measurement deviations were necessary to reduce the candidate set. Since in our case study the faults produced only incipient changes on the measurements, fault detection times were fairly slow since the fault effect has to get large enough before small changes in the state and presence of a fault can be detected. However, for faults in V_1 and V_2, fault detection was fast because these quickly produced significant changes that were easy to detect. Moreover, incipient faults do not cause discontinuous changes in the magnitude of sensor values, and hence, may result in more qualitative ambiguity, causing slower fault isolation than for abrupt faults, which often cause discontinuous changes in the magnitude of measurement values. In each experiment, fault identification always determined the true fault candidate with good accuracy, and in the cases where qualitative fault isolation could not provide a unique candidate, fault identification made it clear which fault was the true fault. For the pump faults, RUL was predicted with high $\overline{\text{RA}}$, ranging above 90%.

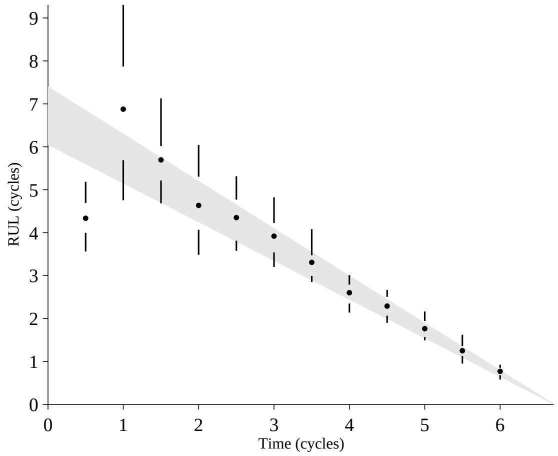

Figure 5: Predicted RUL of pump in the number of loading cycles (1 cycle = 10000 s). The mean is indicated with a dot and confidence intervals for 5% and 95% by lines. The gray cone depicts an accuracy requirement of 10%.

5 CONCLUSIONS

This paper presented an integrated model-based diagnostic and prognostic framework. Our approach makes use of a common modeling paradigm to model both the nominal behavior and fault progression. We demonstrated our approach on a representative propellant loading system, where we diagnosed faults and prognosed the RUL accurately.

While a large body of research exists for diagnostics and prognostics, most approaches focus on either diagnosis or prognosis, but some notable exceptions include (Patrick *et al.*, 2007; Orchard and Vachtsevanos, 2009). However, unlike our approach, these approaches use a single model incorporating all possible faults to estimate states and parameters for different stages of diagnosis and prognosis. For real-world sys-

Table 3: Diagnosis Results

True Fault	True ϕ_f	k_f	Δk_d	Δk_i	Fault Candidates	Estimated ϕ_f	$\overline{\text{RA}}$	Max. No. Fault Identifiers
Nominal	N/A	N/A	∞	∞	$\emptyset$	N/A	N/A	N/A
A_{l1}^+	1.00×10^{-3}	1000	94	94	$\Delta A_{l1} = 1.00 \times 10^3, e = 2.36 \times 10^{-3}$ $\Delta x_3 = 1.39, e = 1.07 \times 10^1$	$\Delta A_{l1} = 1.00 \times 10^{-3}$	N/A	2
A_{l2}^+	1.00×10^{-3}	1000	224	224	$\Delta A_{l2} = 9.98 \times 10^{-4}, e = 2.24 \times 10^{-3}$ $\Delta x_2 = 1.80, e = 9.03$	$\Delta A_{l2} = 9.98 \times 10^{-4}$	N/A	2
x_1^-	-5.00×10^{-1}	1000	0	14	$\Delta x_1 = -5.00 \times 10^{-1}, e = 2.32 \times 10^{-3}$	$\Delta x_1 = -5.00 \times 10^{-1}$	N/A	1
x_2^-	-5.00×10^{-1}	1000	0	13	$\Delta x_2 = -5.00 \times 10^{-1}, e = 2.27 \times 10^{-3}$	$\Delta x_2 = -5.00 \times 10^{-1}$	N/A	1
x_3^+	5.00×10^{-1}	1000	103	111	$\Delta x_3 = 4.99 \times 10^{-1}, e = 2.30 \times 10^{-3}$	$\Delta x_3 = 4.99 \times 10^{-1}$	N/A	1
A_i^-	3.00×10^{-3}	1	933	3116	$w_{A_i} = 3.13 \times 10^{-3}, e = 2.57 \times 10^{-3}$	$w_{A_i} = 3.13 \times 10^{-3}$	96.19	1
r_t^+	8.00×10^{-11}	1	491	491	$w_t = 7.37 \times 10^{-11}, e = 2.72 \times 10^{-3}$	$w_t = 7.37 \times 10^{-11}$	96.75	1
r_r^+	9.00×10^{-11}	1	428	428	$w_r = 9.40 \times 10^{-11}, e = 2.57 \times 10^{-3}$	$w_r = 9.40 \times 10^{-11}$	91.28	1

tems, the large number of states and parameters would lead to scalability issues for these approaches. Moreover, while the system is still nominal, a lot of computational resources are used to estimate faults that are not present. Our approach makes use of multiple single-fault observers to address the efficiency and scalability issues. Also, we save on computational resources by only estimating fault parameters after the fault has detected, and the fault candidates are reduced to a tractable number. Multiple single-fault observers also yield more accurate fault identification than an observer using a model that includes all possible faults.

In future work, we will apply this approach to larger systems, to study the scalability of our diagnosis and prognosis scheme; and expand the capability of this approach to hybrid systems, as well as diagnosis and prognosis of multiple faults. Finally, we will compare our approach with other relevant techniques for integrated diagnosis and prognosis.

REFERENCES

(Arulampalam *et al.*, 2002) M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. on Signal Processing*, 50(2):174–188, 2002.

(Biswas and Mahadevan, 2007) G. Biswas and S. Mahadevan. A hierarchical model-based approach to systems health management. In *Proc. of the 2007 IEEE Aerospace Conf.*, March 2007.

(Daigle and Goebel, 2011) M. Daigle and K. Goebel. Multiple damage progression paths in model-based prognostics. In *Proc. of the 2011 IEEE Aerospace Conf.*, March 2011.

(Daigle *et al.*, 2007) M. J. Daigle, X. D. Koutsoukos, and G. Biswas. Distributed diagnosis in formations of mobile robots. *IEEE Trans. on Robotics*, 23(2):353–369, April 2007.

(Daigle *et al.*, 2010) M. J. Daigle, I. Roychoudhury, G. Biswas, and X. Koutsoukos. A comprehensive diagnosis methodology for complex hybrid systems: A case study on spacecraft power distribution systems. *IEEE Trans. on System, Man, and Cybernetics, Part A*, 4(5):917 – 931, September 2010.

(Goodrich *et al.*, 2009) C. Goodrich, S. Narasimhan, M. Daigle, W. Hatfield, R. Johnson, and B. Brown. Applying model-based diagnosis to a rapid propellant loading system. In *Proc. of the 20th Int. Workshop on Principles of Diagnosis*, pages 147–154, June 2009.

(Hutchings, 1992) I. M. Hutchings. *Tribology: friction and wear of engineering materials*. CRC Press, 1992.

(Kallesøe, 2005) C.S. Kallesøe. *Fault detection and isolation in centrifugal pumps*. PhD thesis, Aalborg University, 2005.

(Lyshevski, 1999) S. E. Lyshevski. *Electromechanical Systems, Electric Machines, and Applied Mechatronics*. CRC, 1999.

(Mosterman and Biswas, 1999) P. J. Mosterman and G. Biswas. Diagnosis of continuous valued systems in transient operating regions. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Trans. on*, 29(6):554 – 565, November 1999.

(Orchard and Vachtsevanos, 2009) M. E. Orchard and G. Vachtsevanos. A particle-filtering approach for on-line fault diagnosis and failure prognosis. *Trans. of the Institute of Measurement and Control*, 31(3/4):221–246, 2009.

(Patrick *et al.*, 2007) R. Patrick, M. E. Orchard, B. Zhang, M. Koelemay, G. Kacprzynski, A. Ferri, and Vachtsevanos G. An integrated approach to helicopter planetary gear fault diagnosis and failure prognosis. In *Proc. of the 42nd Annual Systems Readiness Technology Conf.*, Baltimore, MD, USA, September 2007.

(Roychoudhury, 2009) I. Roychoudhury. *Distributed Diagnosis of Continuous Systems: Global Diagnosis Through Local Analysis*. PhD thesis, Vanderbilt University, 2009.

(Saxena *et al.*, 2010) A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel. Metrics for offline evaluation of prognostic performance. *Int. Journal of Prognostics and Health Management*, 2010.