Engineering Applications of Artificial Intelligence 26 (2013) 446–455

Contents lists available at SciVerse ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A data-based framework for fault detection and diagnostics of non-linear systems with partial state measurement

Niranjan Subrahmanya, Yung C. Shin*

School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, USA

ARTICLE INFO

Article history:
Received 25 August 2011
Received in revised form
30 March 2012
Accepted 3 September 2012
Available online 27 September 2012

Keywords:
Partial state measurement
Fault detection and diagnostics
Non-linear stochastic observer
Recurrent neural networks
Condition monitoring

ABSTRACT

A novel framework based on the use of dynamic neural networks for data-based process monitoring, fault detection and diagnostics of non-linear systems with partial state measurement is presented in this paper. The proposed framework considers the presence of three kinds of states in a generic system model: states that can easily be measured in real time and *in-situ*, states that are difficult to measure online but can be measured offline to generate training data, and states that cannot be measured at all. The motivation for such a categorization of state variables comes from a wide class of problems in the manufacturing and chemical industries, wherein certain states are not measurable without expensive equipments or offline analysis while some other states may not be accessible at all. The framework makes use of a recurrent neural network for modeling the hidden dynamics of the system from available measurements and uses this model along with a non-linear observer to augment the information provided by the measured variables. The performance of the proposed method is verified on a synthetic problem as well as a benchmark simulation problem.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Process monitoring and fault detection methods may broadly be divided into two classes: signal-based methods and model-based methods, and a large number of applications may be found in literature for both the methods (Isermann, 2006; Patton et al., 2000). Signal-based methods (Chen and Liao, 2002, Qin, 2003) generally do not need mathematical models of the system but need data from faulty conditions to perform fault detection and diagnostics. This is desirable in many real world applications as the process may be too complex to model mathematically (as in manufacturing applications) or the effort required in developing a model may not be justifiable economically. Model-based methods (El-Farra and Ghantasala, 2007; Zarei and Poshtan, 2010; Castillo et al., 2012) on the other hand make use of the fact that faults may change the nature of the relationship between the measured inputs and outputs and thus allow the detection of deviations in quantities that are not directly observed. Fault detection may usually be done without the need for data from faulty conditions. Fault diagnosis, however, may, still require data from faulty conditions (Uppal et al., 2006), which is usually difficult to acquire in many applications. Therefore, this paper proposes an approach that tries to retain the benefits of both signal-based and model-based methods by developing a hybrid data and model-based framework that can learn dynamic process models from historical data and diagnose a class of faults without the need for data from faulty conditions or complex physics-based models of the system.

For many industrial systems, approximate data-based dynamic process models can be developed from historical process data using techniques such as auto regressive models with exogenous inputs (ARX) and neural networks. Recurrent Neural Networks (RNN) in particular have been found to be very effective in modeling non-linear dynamic systems (Hou et al., 2007; Lee et al., 2001) and can be used to approximate any discrete dynamic system, which can be represented in the state space form, to any desired degree of accuracy (Jin et al., 1995). This developed data-based model may then be used for fault detection and isolation (FDI).

Model based fault detection and isolation methods have also been called as analytical redundancy methods as they involve the comparison of measured signals with their estimates, based on models subject to the same input condition, to generate residuals. Many model based methods have been proposed in the literature and a short survey of these methods is provided here. Diagnostic information may be extracted from these residuals using simple limit checks or statistical tests (Montgomery, 2008). More robust methods of evaluating residuals including the use of adaptive threshold evaluation (Patton et al., 2000) and non-linear classifiers (Chen and Patton, 1999) have also been proposed. The generation of residuals, however, remains the focus of most model based FDI methods. Residuals can be generated using any of a number of different methods mentioned below. A direct comparison of measured outputs with physics based models (Moskwa et al., 2001; Song et al., 2003; Conatser et al., 2004) or

* Corresponding author. Tel.: +1 765 494 9775.
E-mail address: shin@purdue.edu (Y.C. Shin).

0952-1976/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.engappai.2012.09.004

data based models (Capriglione et al., 2003; Calado et al., 2006; Uppal et al., 2006; Witczak et al., 2006) is the most straight-forward way to generate residuals. Parity relation approaches generate the residual based upon consistency checking on system input and output data over a time window (Gertler, 1997). Parameter estimation approaches directly make use of system identification techniques to isolate changes in critical but unmeasurable system parameters (Isermann, 1993). Observer based methods for deterministic (Edwards et al., 2000; Hou and Patton, 1996; Frank and Ding, 1997) and stochastic systems (Tsai et al., 2007; Kobayashi and Simon, 2006; Li et al., 2005; Xiong et al., 2007) can be used to estimate unmeasured system states/parameters and with suitably designed gain matrices, they can be used to generate residuals which are robust to modeling uncertainties. A bank of dedicated observers can be used to isolate faults for multi-input multi-output systems. The problem of robust model-based FDI in non-linear process systems, especially for actuators, has received significant attention in recent years (El-Farra and Ghantasala, 2007; Zarei and Poshtan, 2010; Castillo et al., 2012). Variations in the structure of the observer banks, such as all input one output, all input all but one output, one input all output, etc. can be used to isolate actuator, sensor and component faults (Isermann, 1997). The parity relation based approach, the observer-based approach and the parameter estimation-based approaches are all related to each other and the correspondence between these approaches may be found in (Chen and Patton, 1999). In this work the observer based approach is used because of the flexibility it affords. In spite of using sophisticated training methods, the use of data-based models, such as the RNN, introduces additional uncertainty regarding model predictions and this should be given due consideration by any model-based fault detection and isolation scheme. While there are a number of non-linear observers that can be considered for the task, such as the Extended Kalman Filter (Reif et al., 1999) or the Unscented Kalman Filter (Julier and Uhlmann, 2004), this paper uses a stochastic non-linear observer called the Adaptive Divided Difference Filter (ADDF), which explicitly accounts for model error and is robust to it (Subrahmanya and Shin, 2009).

Most model-based methods for FDI developed so far assume the availability of a first-principles based model wherein important states of the system are part of the model and may be estimated if necessary. Data-based dynamic models on the other hand may not have interpretable states and if some of the important states of the system need to be estimated on-line (for the purpose of monitoring the process) from input–output measurements, then special care has to be taken to ensure that these states are modeled explicitly. In practice, the instrumentation required to measure all states may not be available and the data available for modeling would include inputs, outputs and selected states. The proposed framework then considers this important practical scenario, where there are three kinds of states in the system model: states that can easily be measured in real time and *in situ*, states that are difficult to measure online but can be measured offline to generate training data, and states that cannot be measured at all. The motivation for such a categorization of state variables comes from a wide class of problems in the manufacturing and chemical industries, wherein certain states (such as surface roughness in manufacturing or intermediate stream compositions in chemical processes) are not measurable without expensive equipments or offline analysis while some other states may not be accessible at all. The goal then is to distinguish faults belonging to three classes—actuator faults (these are assumed to change the influence of an input on the model), component faults (it is assumed that these faults can be detected and diagnosed by monitoring certain states of the system) and sensor faults (these are assumed to affect the measured states). While it is possible that there may be a number of faults in complex systems, which affect multiple elements (inputs, states and outputs), it is believed that the above categorization is still useful to get a general idea of the location and effect of a fault. To the best of the authors' knowledge, this is the first paper considering the combination of a data-based model-based FDI system with this practically important categorization of state variables.

A block diagram describing the architecture of the proposed framework is shown in Fig. 1. The methods used in the three major blocks in Fig. 1 will be described in the following sections. It should be noted that our works on various individual modules in Fig. 1 have been reported elsewhere and the main contribution of this work is the combination of these individual modules and the validation of the entire data-based fault detection and diagnostics scheme. First, the use of recurrent neural networks is proposed for the purpose of learning the dynamics of a system and a suitable structure and training algorithm for the RNN model is given (Subrahmanya and Shin, 2010). A description of the adaptive divided difference filter (ADDF), a robust stochastic non-linear observer for discrete-time systems (Subrahmanya and Shin, 2009), is given next. This is followed by a section on the fault detection and isolation logic for input (actuator), state (component) and output (sensor) faults. Finally a couple of examples, one based on a synthetic state-space model and one based on the DAMADICS simulation benchmark (Bartys et al., 2006), are given to demonstrate the feasibility of the proposed method.

2. System modeling using recurrent neural networks

Although a number of training methods have been proposed for RNNs as mentioned in the introduction, all these methods require a considerable amount of parameter and structure tuning from an experienced user. In order to automate the process of structure and parameter learning for RNNs, the authors recently proposed a constructive training method for RNNs (Subrahmanya and Shin, 2010). This method ensures the stability of an RNN with a single hidden layer throughout the training process as additional nodes are added to its hidden layer. The model structure and training algorithm is presented here. For more details about the properties and performance of the training method the reader is referred to Subrahmanya and Shin (2010).

An RNN can be represented in the discrete state space form with all the measured variables (including the measured states of the original system) as outputs and the hidden node activations as the states. Assume that the discrete state space equation representing the system is

$$x_{k+1} = f(x_k, u_k)$$
$$y_k = h(x_k) \tag{1}$$

where $x_k \in R^n$ is the state vector, $y_k \in R^p$ is the output vector, $u_k \in R^m$ is the input vector and $f(x_k,u_k)$ is the non-linear plant model and $h(x_k)$ is the non-linear observation equation. Using the capability of feed forward networks with a single hidden layer of sigmoidal units to represent any Lipschitz continuous function to an arbitrary degree of accuracy, the functions f and g may be replaced by equivalent feed forward networks as given below, where σ denotes the tanh function

$$f(x_k, u_k) = \mathbf{V}_f \sigma(\mathbf{W}_f x_k + \mathbf{B}_f u_k + \boldsymbol{\theta}_f)$$
$$h(x_k) = \mathbf{V}_h \sigma(\mathbf{W}_h x_k + \boldsymbol{\theta}_h) \tag{2}$$

Here, $\boldsymbol{W}_f$ ($\boldsymbol{W}_h$) and $\boldsymbol{B}_f$ denote weights of a single-hidden layer neural network from the input layer to the hidden layer, $\boldsymbol{\theta}_f$ ($\boldsymbol{\theta}_h$) denotes the biases of the hidden nodes and $\boldsymbol{V}_f$ ($\boldsymbol{V}_h$) denotes the weights from the hidden layer to the output layer to model $f(x_k,u_k)$ ($h(x_k)$). After some manipulation, it may be shown that the dynamics described by (2) can be represented by the system of equations given in (3). Eq. (3) represents a single hidden layer RNN with $\boldsymbol{W}$, $\boldsymbol{B}$, $\boldsymbol{\theta}$ and $\boldsymbol{V}$ as the weights and η_k as the hidden node

Download English Version:

https://daneshyari.com/en/article/381072

Download Persian Version:

https://daneshyari.com/article/381072

Daneshyari.com