DYNAMIC BAYESIAN NETWORK BASED FAULT DIAGNOSIS ON NONLINEAR DYNAMIC SYSTEMS

By

Jiannian Weng

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

May, 2013

Nashville, Tennessee

| Approved: | Date: |
| --- | --- |
| Prof. Gautam Biswas | 03/22/2013 |
| Prof. Sandeep Neema | 03/22/2013 |

**ACKNOWLEDGMENTS**

I would like to say thanks to my advisor, Professor Gautam Biswas for all his support, the instructions and advice he provided me with during this project and for giving me so much freedom to explore and discover interesting research on DBN modeling and probabilistic reasoning. His classes, his extraordinary teaching ability to explain problems, enthusiasm, immense knowledge and weekly meetings have proved to be one of my best learning experiences at Vanderbilt. Although I am not so productive in the research and not be able to survive through the Ph.D. career, Dr. Biswas is always being supportive and encouraging. Lucky for me. I would also like to say thanks to Prof. Sandeep Neema for his advice on the project and served as my committee member, for half year's RA work under META project, for his insightful comments.

I would like to thank Institute of Software Integrated Systems (ISIS) of Vanderbilt University, for providing a wonderful working environment and all the great computing equipments. There are many friends and colleagues I have been pleasure of working with, including brilliant team members Josh Carl, Chetan Kulkarni, Hamed Khorasgani, Daniel Mack and Indranil Roychoudhury (although we never met, much of work in this thesis based on his Ph.D. work).

I would like to thank Prof. Yuan Xue's group, her support during my first year, and her insights and criticisms about my previous work on NCSWT, an integrated modeling and simulation tool for networked control systems. It was fun working with wonderful team members Jia Bai, Xiaowei Li, Fan Qiu, and Wei Yan. I take this opportunity to thank Prof. Xenofon Koutsoukos for his suggestions and insightful discussion on NCSWT. I would also like to thank Prof. Yi Cui, for providing me with the opportunity to be a TA on the Web Development class, and then actually led me to study cloud computing, and gave me the very first thought to the Amazon Web Service.

I am very grateful and I would not have contemplated this road if not for my family. They always be there with me. Although they hardly understand what I'm learning here and what writing thesis is about, even my older brother, they are willing to support any decision I make. Their support is the greatest motivation to finish this thesis.

COMPUTER SCIENCE

DYNAMIC BAYESIAN NETWORKS BASED FAULT DIAGNOSIS ON NONLINEAR DYNAMIC SYSTEMS

JIANNIAN WENG

Thesis under the direction of Professor Gautam Biswas

Fault diagnosis approaches for nonlinear real-world systems play a very important role in maintaining dependable, robust operations of safety-critical systems like aircraft, automobiles, power plants and planetary rovers. They require online tracking functions to monitor system behavior and ensure system operations remain within specified safety limits. It is important that such methods are robust to uncertainties, such as modeling errors, disturbance and measurement noise. In this thesis, we employ a temporal Bayesian technique called Dynamic Bayesian Networks (DBNs) to model nonlinear dynamic systems for uncertain probabilistic reasoning in diagnosis application domains. Within the DBN framework, we develop the modeling scheme, model construction process, and the use of the models to build diagnostic models for online diagnosis. This thesis also performs a preliminary comparison of two particle filter algorithms: generic particle filters (GPF) and auxiliary particle filter (APF). These are commonly used for tracking and estimating the true system behavior. Our approach to diagnosis includes a DBN model based diagnosis framework combining qualitative TRANSCEND scheme and quantitative methods for refining the fault isolation, and using parameter estimation techniques to provide more precise estimates of fault hypotheses. As a proof of concept, we apply this DBN based diagnosis scheme to the Reverse Osmosis (RO) subsystem of the Advanced Water Recovery System (AWRS). Performance of the two particle filter algorithms are compared based on a number of fault scenarios and different levels of noise as well. The results show our DBN-based scheme is effective for fault isolation and identification of complex nonlinear systems.

Approved Prof. Gautam Biswas Date 03/22/2013

**TABLE OF CONTENTS**

## LIST OF TABLES

Table Page

## LIST OF FIGURES

# CHAPTER I

## Introduction

Developing fault diagnosis methods for real-world safety-critical systems, like aircraft, trains, automobiles, power plants, space systems, and planetary rovers is complex because these systems are nonlinear and subject to disturbances and noise that are non-Gaussian (Dearden and Clancy, 2002). These systems require online tracking functions to monitor system behavior and to ensure that system operations do not exceed safety limits. It is important that such methods are robust to uncertainties, such as modeling errors, disturbances and measurement noise (Francisco and Marek, 2003). Model-based methods for diagnosis of such systems use measurements to estimate system state, fault detection methods to determine the occurrence of faults and anomalies, and fault isolation schemes to identify the true fault in the system. Follow up actions can be implemented to maintain system operations and avoid accidents, even human life loss (Weng and Biswas, 2012).

This thesis develops a dynamic Bayesian network (DBN) (Murphy, 2002) based diagnosis methodology for handling uncertainties in tracking system behavior, and diagnosing faults in complex, dynamic systems. A clear and general definition of the diagnostic reasoning problem for dynamic systems operating under uncertainty is formally defined in uncertain domains. Based on the DBN model, two types of particle filters for tracking and estimating dynamic system behaviors, are implemented and discussed. The effectiveness and correctness of this methodology is demonstrated by building a detailed model of the Reverse Osmosis (RO) system of the Water Recovery System (WRS) of the Advanced Life Support System (ALS) (Biswas et al., 2004). The data, nominal and faulty data, is collected from an experiment testbed generated using BDM developed by (Szarka, 2011). Various fault scenarios were created and simulated with our DBN based diagnosis framework. Experimental studies conducted with simulated data are presented, and the effectiveness of the approach is discussed.

### I.1 Motivation

Mission critical dynamic system such as the Water Recovery System (WRS) of the Advanced Life Support Systems (ALS), which was designed to support life for extended duration manned space missions (Duffield and Hanford, 2002) contain a number of interacting subsystems, such as the Biological Water Processing system (BWP), Reverse Osmosis system (RO), Post Processing (PP) and Air Evaporation (AES) (Biswas et al., 2004), that must operate at a high level of autonomy so as not to detract from other mission specific tasks of the crew. This naturally requires a diagnostic framework that can be applied to detect, isolate and identify faults quickly and correctly so as to ensure overall healthy operations. For such an integrated system, the individual components should be robust to a range of fault occurrences working in the hostile environment so that the whole system can have the ability to adapt to changing mission objectives, respond to unplanned events, and even self-tune to work in an acceptable error range (Struss et al., 2010).

Generally speaking, dynamic systems are systems that involve change at all time scales, which are related sets of processes and reservoirs (places where things can reside or forms in which matter or energy exists) through which material or energy flows, characterized by continual change (Bice, 2001). It is very important and meaningful to understand how these system works, especially how they respond to changes. Uncertainties are unavoidable since our models are usually only an approximation to the real system or because of our lack of knowledge, which can be classified into disturbance signals and dynamic perturbations (Lehner and Sadigh, 1991). Especially for our model-based approaches, the system models must be built at the correct level of abstraction, balancing the details needed in the model to make the system diagnosable, while keeping the model complexity low so as not to affect the performance of online diagnosis. Furthermore, the data may be incomplete, ambiguous, erroneous, or imprecise (Gustavo et al., 1996). These uncertain data may adversely affect the system behavior, hence making people even harder to estimate and track the time-dependent system, which differentiate dynamic system from traditional static system.

For present-day, model-based fault diagnosis on dynamic system robust to unlikely events and unanticipated situations has been viewed as the key to maintaining system performance, ensuring system safety, and prolonging system health (Struss et al., 2010). One set of methods, originated from the field of artificial intelligence, system models are formalized as a set of interconnected component models, and a range of algorithms have been developed for localizing and identifying faults in the components. In parallel, the systems dynamic community have developed model-based fault detection and isolation (FDI) approaches that include the parity-space approach (Gertler, 1998), the observer-based approaches (Kabore et al., 1999), and methods based on parameter identification (Ding, 2008). And some other methods are hybrid, and have adopted qualitative schemes, mixed numerical/qualitative models, topological bond graphs, and probabilistic graphical models like Bayesian networks (Russell and Norvig, 2010). In addition, faults could manifest at various locations, and assume a variety of profiles, such as abrupt, incipient, and intermittent, which may or may not cause detectable changes in system behavior. This also requires fault diagnosis methods to be generally applicable to different kinds of faults.

To sum up, several developments over the past 25 years have increased the need for online monitoring and diagnosis in a variety of real-world applications. These drivers include the following: increased needs for performance and safety (particularly for safety-critical systems), increased complexity of systems (with the concomitant increased difficulty of manual supervision), and economic factors, like limiting expensive down-time of plants, reducing maintenance costs, and improving customer satisfaction, and increased uncertain factors coming from measurement and process noise, modeling abstractions and unavoidable errors. Supported by the availability of increased computing power embedded in physical processes, these drivers have led to a high demand and industrial need for supporting and automating effective diagnostic processes. Based on an intuitive and theoretically sound mathematical foundation which generates consistent diagnostic results under uncertainties, probabilistic reasoning techniques are well suited for this purpose. Dynamic Bayesian Networks (DBN) which capture the uncertainties in the system to be diagnosed and relations among system states as well as measurements, are one kind of such probabilistic system models. Once such model created or generated from physical systems, appropriate probabilistic inference approaches like standard Bayesian inference or particle filters could be used to diagnose faults correctly in presence of uncertainties. The inherent difficulties in developing diagnosis models and inference algorithms at an appropriate level of generality lays on two parts: the high costs in building models and diagnostic inference should be made more effective. However, even approximate Bayesian inference schemes can be computationally expensive for huge systems and may suffer from convergence issues.

### I.2 Organization of the thesis

The goal of fault diagnosis is to detect and localize faulty component in a system before the system performance degrades so much that it damages the system, and maybe, its human occupants. The thesis addresses the problem of robust diagnosis in complex, nonlinear systems. It is organized as follows. We begin this thesis by briefly reviewing the past related work on model-based diagnosis of continuous systems, mathematically defining the fault diagnosis problem in complex nonlinear systems with uncertainty and discussing our diagnosis architecture in Chapter 2. Two types of faults, abrupt fault and incipient fault, are also characterized. The qualitative diagnosis scheme of TRANSCEND is then introduced, along with the whole modeling chain, from building bond graph models to deriving temporal causal graphs from the bond graph models. The overall model-based diagnosis architecture is also described.

Chapter 3 presents the temporal Bayesian method for diagnosis using dynamic Bayesian networks. Its relationship with Bayesian network, specific representation, model construction methods, and two ways (generic particle filter and auxiliary particle filter) to do diagnosis reasoning are discussed. The two-tank system and an electrical system are used as an example to make the theory more understandable.

Chapter 4 presents a case study, the Reverse Osmosis system, part of the Advanced Water Recovery system, to demonstrate the effectiveness of DBNs based fault diagnosis and compare the performance of these two particle filters in tracking and estimating state under nominal and faulty conditions.

The discussion and conclusion of this thesis are presented in Chapter 5.

# CHAPTER II

## Background

Traditional approaches to the diagnosis are based on a predetermined set of allowable faults, and they typically fall short in both the performance and diagnostic resolution as the complexity of system and the number of possible faults increase (Chen, 1994). Modern model-based diagnosis methods employ a model that is derived from system's structure and behavior in order to establish the cause of system malfunctions (Ding, 2008). While a number of different model-based fault diagnosis algorithms have been proposed in the past decades, probabilistic reasoning based on Bayes networks from the AI community has been adapted as a generalized method for tracking the nominal and faulty behavior of nonlinear dynamic systems in uncertain environments. For uncertainty, we can deal with it in two ways: extensionally and intensionally (Mihajlovic and Petkovic, 2001). Extensional systems (also called rule-based systems) are computationally efficient but their uncertainty measures are semantically weak. On the contrary, intensional systems are generally computationally expensive and semantically strong. By assigning random variables to represent events and objects in the world, the current state of the world can be modeled and analyzed according to their joint probabilities. We begin this chapter by summarizing previous work done by many excellent researchers before focusing on different probabilistic models for this purpose according to domain of interest, such as, Bayesian reasoning, evidence theory, robust statics, and recursive operators.

### II.1 Related work in Model-based Diagnosis of Dynamic Systems

Model-based diagnosis can be considered as search for system model consistent with observations, while consistency is checked with logical and algebraic methods (Chen, 1994). Recently, there is an increasing interest in both research and applications of model-based diagnosis mainly due to their great advantages and well-founded theoretical backgrounds. This approach is based on an explicit system model applied for diagnostic inference. In most cases, the model is component-oriented, just like dynamic Bayesian network model we propose and employ in this thesis. For each type of component, it includes: a list of its variables (interface, internal or state variables, parameters), as well as its modes of behaviors (correct and fault modes). The behavior could describe a set of relations (algebraic and integral). Both the nominal and faulty behaviors can then be described exploiting different modeling assumptions. There are many variations on such model-based diagnosis. For example, a model may be qualitative (often based on cause/effect models) or quantitative (based on numbers and equations); static or dynamic (if it is evolving over time); non-causal models or causal (if it captures cause/effect information); deterministic or probabilistic models (where incomplete and/or uncertain information is represented using numeric information) (Console and Dressler, 1999). Bayesian network has proved to be a very good way of dealing with uncertainty of dynamic systems in diagnosis application domain.

Dynamic Bayesian network(DBN) extends static Bayesian network(BN) by introducing the notion of time, namely, by adding time slices and specifying transition probabilities between these slices. Based on Markov assumption, future system states are independent of the past states of the system once the present states are known. Hence, a first-order DBN is usually defined as two-time slice BN (2TBN), where the intra-slice dependencies are described by a static BN and inter-slice dependencies describe the transition probabilities. Diagnosis techniques based on DBN become more and more important in probabilistic fault diagnosis techniques for dynamic system. Because of the computational complexity of exact inference algorithms based on DBN increases quite quickly with the number of nodes, the research community then convert to approximate algorithms which include The Boyen-Koller (BK) algorithm (Boyen and Koller, 1998), the Factored Frontier (FF) algorithm (Zweig, 1996), the interface algorithm (Darwiche, 2001) and Particle Filter (PF) algorithm (Arulampalam et al., 2002). Most of them decrease the computational complexity at the cost of sacrificing diagnosis accuracy.

(Lerner et al., 2000) employs DBN representing both nominal and various faulty system behaviors including burst faults, measurement errors and gradual drifts, to track and diagnose complex systems with mixtures of discrete and continuous variables. It focus on five-tank systems that are composed of several weakly interacting subsystems and future observations are used to help determine likely fault candidates to keep unchanged and unlikely ones to be collapsed more aggressively.

(Boyen and Koller, 1998) use DBN to monitor dynamic system current status and future trajectory, and demonstrate how the additional structure of a DBN can be used to design approximation scheme, improving its performance significantly. The reasoning algorithm proposed maintains an approximate belief state with compact representation, which also propagate from one time slice to the next. This method guarantees that the error from approximation do not accumulate. It is validated by applying to water purifying process and the BAT (Bayesian Automated Taxi) network (Forbes et al., 1995).

(Kawahara et al., 2005) applied DBN into diagnosis for spacecraft. The DBN are initially generated from prior knowledge, then modified or partly re-constructed by statistical learning with operation data. It shows that even in complicated fault cases DBN based approach can detect anomaly and make a short list of the fault positions. However, due to the incompleteness of system's observability and the inacuracy of DBN's representation power, it could be difficult to specify the faults completely.

(Roychoudhury et al., 2010) presents DBN-based distributed diagnosis scheme, where each distributed diagnoser generates globally correct diagnosis results without a centralized coordinator by communicating a minimal number of measurements to decrease the computational complexity. Each local diagnoser guarantees globally correct diagnosis results and experiment results on an electrical circuit demonstrate the efficacy of their diagnosis scheme. (Alonso-Gonzalez et al., 2010) also proposes decomposing a system with Possible Conflicts (PCs) and afterwards, building a DBN factor from each resultant PC to distribute the diagnosis process and reducing the heavy computational burden.

Apart from diagnosis domain, DBN can be also applied for information fusion where the decisions must be made efficiently from dynamically available information (Zhang and Ji, 2006), for bioinformatics using perturbed gene expression data (Dojer et al., 2006), for identifying gene regulatory networks (Zou and Conzen, 2005), for gesture interaction, audio-video conversation, football game (Jebara, 2005), for vehicle classification in video (Kafai and Bhanu, 2011), and so on. It is a very active research topic that becomes more and more popular with a great help of theoretical foundation from probabilistic theory.

### II.2 Problem statement

Consider the state equations of a continuous dynamic system (Gustavo et al., 1996):

$$\dot{x(t)} = A(t)x(t) + B(t)u(t) + Ed(t) + Kf(t)$$

$$y(t) = C(t)x(t) + D(t)u(t) + Fd(t) + Gf(t),$$

and the coefficients $A$, $B$, $C$, $D$ are time-varying to denote the nonlinearity of the system. The term $Ed$ models the unknown inputs to the dynamic process, $Kf$ represents the component faults, $Fd$ the unknown input to the sensor and $Gf$ the possible sensor faults.

Definition: A *fault* is an unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable, usual, standard condition (de Kleer and Kurien, 2003).

Fault is quite different from *failure* which suggests a complete breakdown of a system or a component, and *malfunction* which means the inability of the components to accomplish its function. The faults are principally reflected in changes of $A$, $B$, and $C$, as well as modeling errors, are considered by $f$ and $d$ associated with proper choices of $E$, $F$, $G$, $K$. While these matrices are usually given, the modes (i.e. evolutions) of $f$ and $d$ are generally be considered unknown.

*A diagnostic problem* is defined as detection and identification of the fault. Consider the dynamic system above with a known nominal model. Given the actual input $u(t)$, and the measurement $y(t)$, suppose that a residual vector $r(t)$ exists that carries information about some faults. The detection problem is to find a way that generate $r(t)$ when the fault has occurred, under following conditions: (1) the mode (time evolution) of the fault is unknown (disturbance and dynamic perturbations); (2) the mathematical model is uncertain

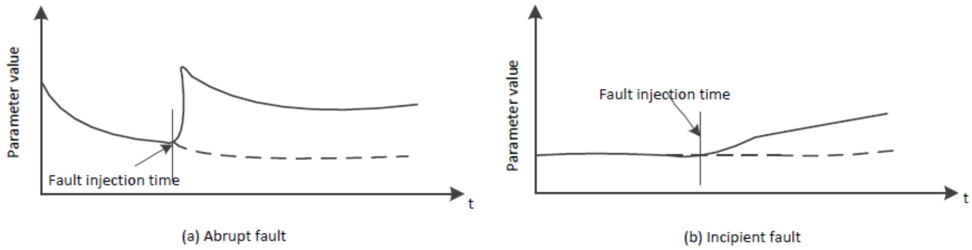


Figure II.1: Fault Profile

(modeling error or our lack of knowledge); (3) the residual generation has to be performed within a specified time. The diagnosis reasoning based on Bayesian methods is to compute the probability of each variable (fault) given other variables's value (measurements) known. Based on Bayes' rule, the problem could be converted into computing the joint probability distribution:

$$P(X_{0:t}, E_{1:t}) = P(X_0) \prod_{i=1}^{t} P(x_i|x_{i-1}) P(E_i|X_i),$$

where $X_{0:t}$ represents states from time 0 to time $t$, $E_{1:t}$ means observations from time 1 to time $t$, and $x_i$ denotes the system state at time point $i$.

Our approach is designed to classify and estimate the fault magnitude of two different fault types: (1) abrupt fault and (2) incipient fault based on a sequence measurements made on the system.

### II.2.1 Abrupt Fault

An abrupt fault, shown in Figure II.1(a), is defined as an instantaneous but fixed change (increase or decrease) in a component parameter value *p(t)*. In reality, no fault is instantaneous, but we approximate changes where the change happens much faster than the sampling rate as an abrupt fault. Note that an abrupt fault for a linear element results in a constant value change, i.e., a parameter value changes from $p$ to $p_f$. For a nonlinear element, the magnitude change in the fault can be modeled as a bias term.

$$p'(t) = \begin{cases} p(t) & \text{t} < t_f \\ p(t) + b(t) & \text{t} \geq t_f, \end{cases} \tag{II.1}$$

where $t_f$ is the fault time of injection, *b(t)* is the fixed bias value, and *p(t)* is the changing measurement in the dynamic system. Consider the circuit system shown in paper (Weng and Biswas, 2012), before fault injection, $R_1$ evolves as $p(t)$, and after injection of fault, there is a constant persistent bias term as $b(t)$, which could also be represented as $\Delta_p^a \times p(t)$. It characterizes a very fast change, i.e. the rate of change is much faster than the dynamics of the system in the system parameter, $p(t)$.

### II.2.2 Incipient Fault

An incipient fault, shown in Figure II.1(b), is often approximated as a linear change (gradual increase or decrease) in the parameter value. Since the incipient fault manifests as a slow gradual change in a parameter, the incipient faults can be approximated by a linear, additive, drift term, $d(t)$, with a constant slope. For example, the slope of the nonlinear resistor in this thesis could be changed to a larger value to model an incipient fault. The mathematical model for the incipient faults is shown below.

$$p'(t) = \begin{cases} p(t) & \text{t} \leq t_f \\ p(t) + c \times (t - t_f) & \text{t} > t_f, \end{cases} \tag{II.2}$$

where $c$ is the constant slope, $p(t)$ represents the nominal parameter value as above and $c \times (t - t_f)$ is the drift function $d(t)$, $t_f$ is the injection time of incipient fault.

Probabilistic reasoning schemes are now used extensively as part of diagnosis algorithms. Probability theory provides mathematically sound reasoning mechanisms based on a numerical degree of belief (between 0 and 1) associated with hypotheses and measurements (i.e. evidences) in a diagnostic scheme. The fundamental problem we seek to solve in a probabilistic diagnosis is to determine the chance of a particular fault occurring given the observed systems. This question, however, is counterintuitive, since our knowledge about the real world is causal. In other words, domain experts usually have a fairly good intuition about the chances of seeing a particular symptom given a fault in the system, e.g. the chances of having a headache if someone having a fever. However, trying to ascertain the chances of the fault happening given a particular effect, e.g. the chances of someone having a fever given he has a headache, is somewhat counter intuitive, and the precise question we ask in a diagnosis problem. In general, Bayes' theorem provides the fundamental mechanism for diagnosing faults in the presence of uncertainty, by relating symptoms to faults. For example, assuming *Symptom* and *Fault* are two random variables, the posterior probability of *Fault* given *Symptom*, $P(Fault|Symptom)$ can be ascertained from "intuitive", causal information such as $P(Symptom|Fault)$, and prior probabilities $P(Fault)$ and $P(Symptom)$ as follows:

$$P(Fault|Symptom) = \frac{P(Symptom|Fault)P(Fault)}{P(Symptom)}$$

If there are $n$ *Symptom* variables,

$$P(Fault|Symptom_1,...,Symptom_n) = \frac{P(Symptom_1,...,Symptom_n|Fault)P(Fault)}{P(Symptom_1,...,Symptom_n)}$$

To calculate the conditional probability on the numerator and assume that the single hypothesis directly influences the evidence, we need to get the full joint probability distribution as follows:

$$P(Fault,Symptom_1,...,Symptom_n) = P(Fault)\prod_{t=1} nP(Symptom_t|Fault)$$

We can see how the probabilities of different hypothesis are updated as more evidence is available. The key is to compute the joint probability distribution. Although the assumption of each evidence variable is conditionally independent from other evidence variables given the hypothesis reduces the computational complexity and the need for a large number of probability values, it is too strict and may not be correct always. Several graphical models like Dynamic Bayesian networks (DBN) which model the system uncertainty and graphically represent the efficient factorizations of the joint probability distributions over a set of variables can behave a correct and efficient inference without such causal dependencies between variables. It is possible because these models capture the multiple causal dependencies, as well as, the independence between different random variables.

### II.3 Previous work on Qualitative Diagnosis Approaches

This section briefly reviews the observer-based TRANSCEND continuous diagnosis scheme (Mosterman and Biswas, 1999), and describes a chain of modeling steps that we employ to build our model-based fault diagnosis methodology. We start with the Bond Graph (BG) modeling framework that forms the core of our system modeling approach and both the state-space equations required by the observers, as well as the Temporal Causal Graph (TCG) for qualitative analysis are automatically derived from these bond graph models. Through this section, we will use a nonlinear electrical system for example, shown as Figure II.2.

#### II.3.1 Bond Graph

Bond graphs are domain-independent, energy based topological models that capture energy exchange pathways in physical process and accommodate nonlinear behaviors (Broenink, 1999). They allow for physical system modeling from first principles, and encode causal and temporal information that are helpful in fault isolation. At the Institute for Software Integrated Systems (ISIS), we have also developed the BDM paradigm (Szarka, 2011) that allow for explicit parametrized representation of sensors and actuators in the system in bond graph modeling language.

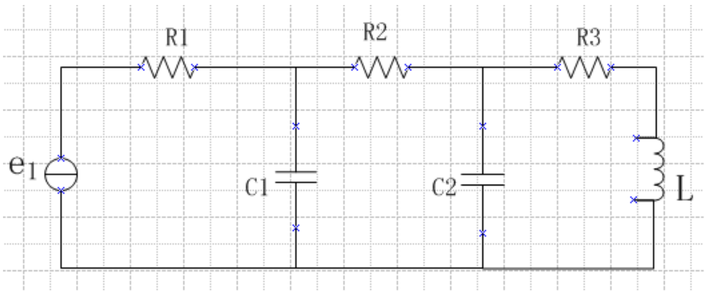


Figure II.2: Third-order Electrical System

Bond graphs define dynamic behavior in terms of the energy exchange between the components of the system. Two generic variables, effort $e$, and flow $f$, define the rate of energy flow in the components of the system. The BG modeling language allows for multi-domain modeling in a common framework. Hence, in the electrical domain, $e$ is defined as voltage and in the mechanical domain it represents force. Similarly, $f$ represents current in the electrical domain, and velocity in the mechanical domain. And pressure difference and volumetric (or mass) flow rate, respectively in the hydraulic domain. The primitive elements associated with nodes are: (1) energy storage elements ($C$ and $I$); (2) energy dissipating elements ($R$); (3) idealized energy transformation elements (transformers $TF$ and gyrators $GY$); and (4) energy source elements ($S_f$ and $S_e$). They are connected in models by two ideal junction elements: $0-$ and $1-$ junctions, based on the conservation of energy and continuity of power. The edges in the directed graph are called *bonds*, which denotes an ideal energy flow between two connected submodels and are drawn as half arrows ($\rightharpoonup$). Each bond specified by a bond number has an associated "across" effort variable $e$ and "through" flow variable $f$ variables, and $e \times f$ denotes the rate of energy transfer through the bond. The topological structure of a bond graph model provides implicit information about the computational causality and dependence of the variables associated with the bonds and components of the model. Causality is denoted by a causal stroke on one end of a bond, with the BG element near the causal stroke imposing flow on the BG element away from the causal stroke. There is a well-defined procedure called SCAP (Dijk, 1994) for assigning causal directions to the bonds in a model, and the resulting bond graph model is called the causal bond graph. With such causality in the BG, it not only allows us to generate the computational forms of BG dynamics (state-space equations or block diagrams), but also helps in determining other important information about the system from its BG, such as the physical validity of the BG model and system observability.

Note that different from the primitive BG elements, some of the BG elements may be algebraic functions of other system variables, or even external signals, are called *Modulated elements*, that can be used to model nonlinearities in a BG and also capture time-varying input to the physical system. Graphically, the signal links

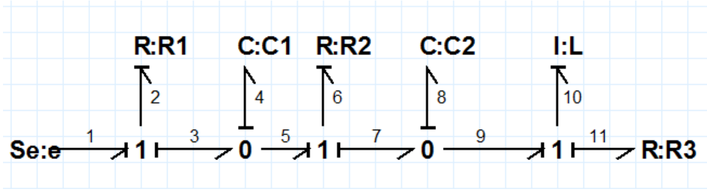


Figure II.3: Bond Graph of Example Electrical System

start from the internal or external components and point to the modulated elements, drawn as full arrows (i.e. $\rightarrow$). They all have a prefix 'M' added to their component names, e.g. MR:$R$ denotes a modulated resistors. We can see an example of nonlinear system modeled as BG in the case study at Chapter 5.

*Example*: Figure II.3 shows the causal BG of a nonlinear electrical system, including three resistors ($R_1$, $R_2$, $R_3$) as dissipative elements, two capacitors ($C_1$, $C_2$) and one inductor ($L$) as energy storage elements, and one ideal voltage source ($V_e$). The series and parallel sections of the circuit model are implemented as 1- and 0- junctions, respectively. The effort variables, $e_4$ and $e_8$ associated with the two capacitor voltages, and the flow variable, $f_{10}$ associated with the inductor current form the state variables of the system as defined using the bond graph convention. The current flowing through two resistors, $f_2$ and $f_6$, voltage on $R_3$ could be detected or measured using some equipment or sensors.

### II.3.2 Temporal Causal Graph

The TCG (Mosterman and Biswas, 1999) can be derived automatically from a causal bond graph, which shows the relations between efforts and flows in the BG and explicitly incorporates the cause and effect relations between fault components and measurements. TCG captures dependencies (algebraic and temporal) between system variables as a causal structure. In the TCG, nodes are effort and flow variables. The direction and type of interaction between nodes are denoted as edges. Labels -1, +1, and =, on the links imply inverse, direct and equality relations between corresponding variables. Besides, edges associated with a component represent the component's constituent relation, such as $\frac{1}{R}$ corresponding to a resistive element and $\frac{1}{C}dt$ denoting flow-to-effort relation for a capacitor in integral causality. Temporal relations in the TCG are associated with the energy storage elements, i.e., $I$ and $C$. All other relations in the TCG, e.g., the voltage-current relations imposed by the resistors and the idealized 1- and 0- junction relations are algebraic. The causal information in TCG allows the deviations of measurements from nominal to be mapped on to possible parameter deviations, and also predict qualitatively the effect each of the parameter deviations would have on the measurements.

The TCG captures the causality of physical effects in the system, and retains the dynamics expressed in

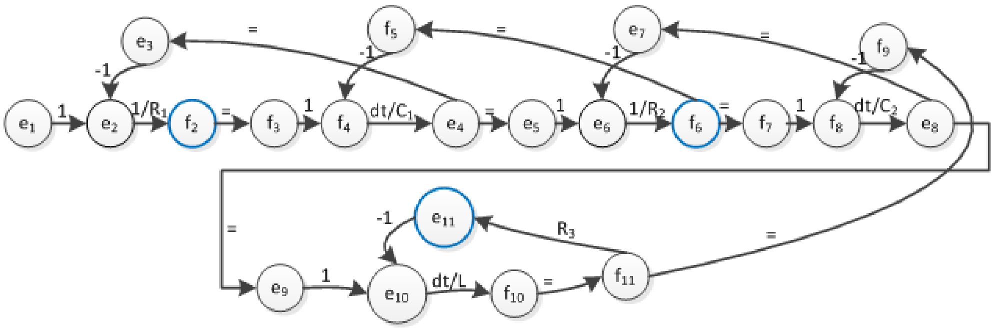


Figure II.4: Temporal Causal Graph of the Electrical System

the bond graph model. In effect, it specifies the signal flow graph, albeit in a form where each edge relation contains at most one component parameter value. Essentially, TCG is a signal flow graph whose vertices correspond to the effort and flow variables of the BG, and the edge denotes the causal dependencies between these system variables. Since the topological structure of the BG and properties of its constituent elements imply inherent causal relations between system variables, TCG can be used to analyze cause-effect from observed behavior deviations to the changing parameter in system components. TCG is the key for fault isolation in the TRANSCEND framework to generate the fault signatures, and then decrease the number of possible faults by actual measurements. And it could also be used to generate dynamic Bayesian network models.

*Example*: Figure II.4 shows the TCG of the third-order electrical system, whose bond graph is shown in Figure II.3. The nodes $e_i$ and $f_i$ in the TCG correspond to effort and flow variables of bond $i$. For example, $e_1$ correspond to bond 1, voltage value of the ideal voltage source. As explained above, under conventional integral causality, $C_1$ imposes effort on its adjacent 0-junction, and hence the edge $f_4 \xrightarrow{dt/C_1} e_4$ is drawn in the TCG. $dt$ label represents integration. Similarly, the edge $f_8 \xrightarrow{dt/C_2} e_8$ is drawn for capacitor $C_2$, $e_{10} \xrightarrow{dt/L} f_{10}$ is for inductor $I$. Resistor $R_1$ relates $e_2$ and $f_2$ according to relation $e_2 = R_1 f_2$. Hence, we have edge $e_2 \xrightarrow{1/R_1} f_2$. At the first 0-junction, bond 4 is the determining bond. Hence, we have $e_4$ = $e_3$ = $e_5$. Therefore, in TCG, we have $e_4 \xrightarrow{=} e_3 \xrightarrow{=} e_5$. At the first 1-junction, we also have $e_2$ = $e_1$ - $e_3$. Hence we have two edges $e_1 \xrightarrow{1} e_2$ and $e_3 \xrightarrow{-1} e_2$. Noted that we don't need to explicitly specify the signal link that connect to the measurement variables in the TCG. Usually the edge labels for these are '=', representing equality. However, it would be better to draw them in the TCG figure if they are functions of TCG elements.

### II.3.3 Qualitative Fault Isolation

Ideally, non-zero residual implies a fault is detected. In order to accommodate uncertainties due to measurement noise and modeling errors, the framework here employ a statistical Z-test to establish if the measurement residuals are statistically significant so as to avoid false alarms but retain the sensitivity of detection.

Once the fault detected, we stop the observer, use the system model to simulate the system and qualitative fault isolation scheme is triggered to generate initial fault hypothesis and refine these hypothesis according to the updated observations. It consists of three steps: (i) Feature Detection, (ii) hypothesis generation (iii) hypothesis refinement. The key idea is that analyze the transients in the measurements caused by faults, and compare the expected deviation of measurements from nominal with the actual observed deviations, represented qualitatively using symbols. Now we present each step in detail below.

*Feature Detection*: Individual signal features are the prime discriminating factor between competing fault hypotheses. Two features are extracted from each measurement residual to denote how the residual changes over time: -, 0, and + symbols, representing below, at, or above nominal values, respectively. The feature consists an ordered pair of symbols. The first one capture the magnitude, while the second one is the slope measurements. There are many specialized algorithms used to derive other useful features from signals in a qualitative framework. Usually, we can have a third general feature called steady state to aid the fault isolation process, since most physical system will eventually return to a steady state due to the dissipative effects.

*Hypothesis generation*: generate possible faults that could explain the measurement deviations observed so far. For every recorded discrepancy between measurement and nominal value a backward propagation algorithm is invoked on the TCG to implicate component parameters. The algorithm propagates observed deviant values backward along the directed edges of the TCG and consistent - and + deviation labels are assigned sequentially to vertices along the path if they do not have one.

*Hypothesis refinement*: if the fault signatures of the generated hypothesis is not consistent with the observed symbol for the updated measurement, the fault hypothesis is dropped. By propagating in the forward direction along the TCG, fault signatures of that fault hypothesis could be generated, which represent the possible effects of the hypothesized faults on observable measurements at the point of failure. This refinement process is continued given the updating measurement till the number of fault hypothesis is refined to a very small number, or even converge to one.

*Example*: we continue with the third-order nonlinear electrical system presented above. If the current through $R_2$ increases gradually, there can be possible explanations for this fault, the increase in resistance $R_2$ and $R_3$, i.e., $R_2^{+i}$ and $R_3^{+i}$ or the degradation in capacitor $C_2$, i.e., $C_2^{-i}$. However, as time evolves, if we observe that the voltage at $R_2$ decreases instead of increasing, then $R_2^{+i}$ is considered very unlikely. Thus, this fault hypothesis could be dropped from the fault candidate set. As more measurement deviations are observed, we can come up with a consistent set of fault hypotheses explaining the deviations. For each fault hypothesis, we propagate in the forward direction along the temporal causal graph and generate fault signature table, the symbolic representation of the possible effects of the hypothesized faults on the observable measurements at the point of failure. Comparing updated measurements deviation from nominal behavior with the fault signatures of the generated fault hypothesis, if it is inconsistent, the fault is dropped. See (Mosterman and Biswas, 1999) and (Roychoudhury, 2009) for more details on progressive monitoring.

### II.4 Model-based Diagnosis Architecture using Dynamic Bayesian Networks

The TRANSCEND qualitative framework always suffers from the ambiguity problem, i.e. the ability to uniquely isolate the true fault from a set of hypotheses due to the lack of discriminatory ability of the qualitative fault signatures. In this section, we introduce our model-based diagnosis framework using Dynamic Bayesian Networks (DBN) to produce more precise diagnoses, and can be made to be more robust.

Figure II.5 shows the computational architecture of our Bayesian diagnosis scheme combined with qualitative framework. We start with the DBN nominal model generated automatically from its TCG for tracking the system nominal behavior. The difference between nominal measurement estimates and the actual observations, defines the residual signals that is then used in fault detectors to detect statistically significant non-zero residual values.

Once a fault detected, the qualitative TRANSCEND scheme is triggered and possible fault hypotheses that could explain the observed measurement deviations are generated. With the hypothesis refinement, the hypothesis set is reduced to a number that is less than a user-specified lower bound, or the fault hypotheses set cannot be reduced any further. At this point, the quantitative fault hypothesis refinement and identification scheme is invoked to identify the true fault hypotheses.

For each fault hypothesis left from qualitative refinement process, we need to generate a faulty DBN, typically done by modifying the nominal DBN model to include the faulty parameter as a stochastic variable in DBN. Since in this thesis we make the single fault assumption, each DBN-based observer is then invoked to track and estimate the observed measurement values using a particle filter scheme. If the estimated measurements significantly deviates from the observed actual faulty measurements, that fault hypothesis is inconsistent and would be dropped. Only one faulty-DBN will produce the values that converge to the observed faulty measurement values. That is the true fault we need. Besides, along with tracking the system behavior, we could estimate the fault parameter changing over time.

### II.5 Summary

In this chapter, we presented previous research on model-based diagnosis schemes, traditional qualitative schemes and probabilistic schemes. They exemplify different methods for handling uncertainties in nonlinear dynamic systems. However, they all have their own individual limitations, for example, less discrimination power for the qualitative method. In addition to kinds of modeling method, various sampling methods have

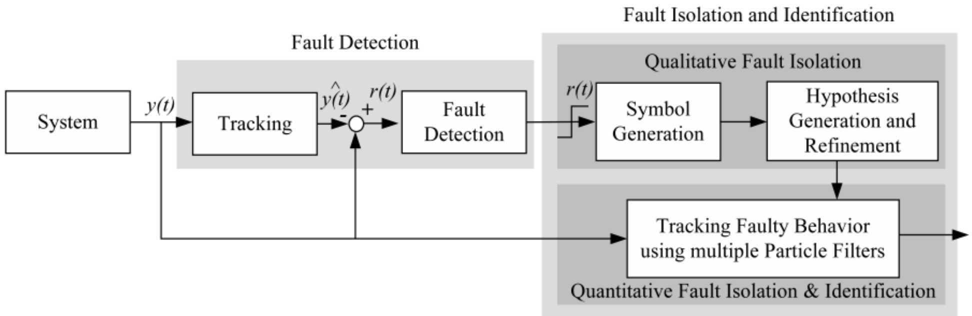


Figure II.5: DBN-model based Diagnosis Architecture

also been proposed to maintain the present belief state. We focus on the development of particle filters.

It is well understood that uncertainties are unavoidable in a real dynamic system. Coming from disturbance signals and dynamic perturbations, the dynamic process suffers from the input and output disturbance. Besides, for a model-based approach, a mathematical model of any real system is always just an approximation of the true, physical reality of the system dynamics, no matter how close the model with the system. Not to mention people's lack of knowledge on the real dynamic systems. These modeling errors may adversely affect the diagnosis results of a system. Probabilistic diagnosis of dynamic systems under uncertainties is mathematically defined by explicitly modeling and reasoning with the process and measurement noise. In probability theory, we consider the system variable to be a random variable, assume a distribution about each parameter and system variable, and Bayesian reasoning approaches could be used to infer correct and accurate diagnosis results in terms of probabilistic distributions in the presence of uncertainties.

Other than system unavoidable uncertainty, we also discuss two types of unwanted changes, namely faults that cause deviations from expected system behavior, which then affect system performance. It is quite different from failures, complete failures that break the whole system down. The faults in this thesis degrade system performance but will not result in a complete shut down of the system functionality. Take electrical system for example, failure may means part of circuit broken causing no current flowing through $R_1$, while faults may means gradually degradation on capacitor or increasing resistance due to the increasing temperature. We adopt the terminology used in the diagnostic domain, such as abrupt fault and incipient fault to present the different concepts in the remainder of this thesis.

With all the fundamental diagnostic problem statement and fault profile presented, we introduce a modeling chain that we employ to build our model-based fault diagnosis methodology. We start with the Bond Graph modeling framework that forms the core of our system modeling approach and both the state-space equations required by the observers, as well as the Temporal Causal Graph for qualitative analysis are automatically derived from these bond graph models. Previous work done by (Mosterman and Biswas, 1999) and (Roychoudhury, 2009) on the qualitative diagnosis framework is briefly described with an example of electrical system. We reviewed how to generate fault hypothesis and refine hypotheses based on the fault signature table. Then we present our computational architecture of combined qualitative-quantitative Bayesian diagnosis scheme. DBN-observer is used to track nominal system behavior and TRANSCEND here is used to generate and refine fault hypothesis. For each remaining fault hypothesis, a faulty DBN is generated by modifying the nominal DBN model to include the faulty parameter as a stochastic variable in the DBN, which will be described in the next chapter. Particle filter scheme is then adopted to estimate states using DBNs. Under single fault assumption, a separate DBN-based observer is then invoked for each fault hypothesis model to track the observed measurement values using a particle filter scheme. Only the particle filter estimator that uses the true fault model produces measurement value estimates that converge to the observed faulty measurement values, while others will significantly deviates from their corresponding components. In the next chapter, we will focus on DBN-based diagnosis method, like its model representation, model construction, and the reasoning methods.

# CHAPTER III

## DBN for diagnosis on Dynamic Systems

Qualitative reasoning schemes such as TRANSCEND and traditional methods such as (Chen, 1994), may mitigate the diagnosis problems by reducing the number of fault hypotheses. However, they cannot precisely represent the time-evolving faulty system behavior and their reasoning framework can lead to ambiguity problems with no ability to distinguish between sets of fault hypotheses when system state space is very large. Even if they are quantitative methods like Kalman Filter (Kalman, 1960), they also have their limitations, such that KF can only applied to linear Gaussian systems. Unfortunately, in the diagnosis reasoning domain application, switching of fault mode can always introduce discontinuous jump from one continuous behavior to another continuous behavior. In this chapter, we will describe a general approach called dynamic Bayesian network (DBN) (Lerner et al., 2000) for diagnosis of complex nonlinear systems.

### III.1 Introduction

As we discussed before, in order to deal with system uncertainty, we could model the current state of the world and weight the states according to the full joint probabilities. Bayesian networks (BN) bring the most appropriate representation of relative influences among the real world facts. As for a temporal point of view, i.e., time-varying patterns (a sample realization of stochastic process consisting of a set of observations made sequentially over time), BN brought a different approach in attempt to model events that include time-series modeling. This new tool is known as the Dynamic Bayesian Network (DBN).

DBNs model systems that are dynamically changing or evolving over time. It enables users to monitor and update the system as time proceeds, and even predict subsequent behavior of the system. Usually, DBN are defined as special case of singly connected BN aimed at time series modeling. These temporal connections are between time slices, that incorporate conditional probabilities between variables. The state variables do not need to be directly observable. They could influence some other variables directly measurable or calculable. In DBNs, each state at one time instance may depend on one or more states at the previous time instance or/and on some states in the same time instance.

The DBN diagnosis model first proposed in (Lerner et al., 2000) includes all the possible faults (single and multiple) in the system. The number of possible faults can be really large in complex systems making the tracking and estimation process computational intractable in the diagnosis framework. In this thesis, we derive a faulty DBN for each corresponding fault hypothesis so as to improve the computational efficiency and address the tractability and accuracy issues when using online PF approaches for monitoring and estimation in the nonlinear dynamic systems. In the following section, we will describe the general DBN model representation, construction and reasoning methods for helping users find the right fault buried inside system's components, for system administrator monitoring the system real-time behavior while maintain the system under normal healthy working conditions.

### III.2 Model Representation

DBNs extend the Bayesian network formalism by providing an explicit discrete temporal dimension. We could represent DBNs into two parts: Qualitative and Quantitative parts.

(1) *Qualitative level*: Assume that the system is modeled evolving in discrete time steps, each time slices contains a set of (time-indexed) random variables, $Z$, which can be real value or discrete value. According to paper (Lerner et al., 2000), we denote the discrete variables as $D_t \subseteq Z_t$, the observable (measurement) part of continuous variable as $Y_t \subseteq Z_t$, and the remain unobservable state variable as $X_t \subseteq Z_t$. Therefore, DBNs at each time slice is represented as $DBN = \{D,Y,X\}$. Sometimes, there will be $U_t \subseteq Y_t$ as the input control variables. The key idea for DBNs is to represent the conditional probability distribution $P(Z_{t+1}|Z_{0:t})$, which includes state transitional model $P(X_t|X_{0:t-1})$. Two assumptions hold. Markov assumption states that the current state only depends on a finite fixed number of previous states, which is used to solve the problem of unbounded set of $X_{0:t-1}$, considering first-order Markov assumption,

$$P(X_t|X_{0:t-1}) = P(X_t|X_{t-1})$$

Stationary process assumption requires that the process of change is governed by law that do not themselves change over time, that is, state changes but conditional dependence relationships doesn't change. Thus, we have our sensor model as:

$$P(Y_t|X_{0:t},Y_{0:t-1}) = P(Y_t|X_t)$$

Therefore, qualitatively, DBNs can be represented as two time slices BN. The nodes are random variables in two consecutive time slices: $Z_t$ and $Z_{t+1}$. Edges capture the direct dependence relations between two nodes it connects, with inter-slice edge modeling the system dynamics (temporal relation) and intra-slice edge modeling instantaneous relation (algebraic). Consider the two tank system that models a chemical process which is commonly used in the fault diagnostic domain for an example (Lerner et al., 2000), Figure III.1 its DBN model. It has the following variables at time $t$: $X_t = \{P1_t, P2_t\}$, the pressures at the bottom of tanks 1 and 2, respectively. There is an input variable $U_t = \{f_{in}\}$, the flow into tank 1, and $Y_t = \{F10_t, F12_t, F20_t\}$, the outflow from tank 1 and tank2 respectively, and the flow between tank 1 and 2. The across-time model includes five links, $P1_t \rightarrow P1_{t+1}$, $P2_t \rightarrow P2_{t+1}$, $P1_t \rightarrow P2_{t+1}$, $P2_t \rightarrow P1_{t+1}$, and $F_{in} \rightarrow P1_{t+1}$. These links

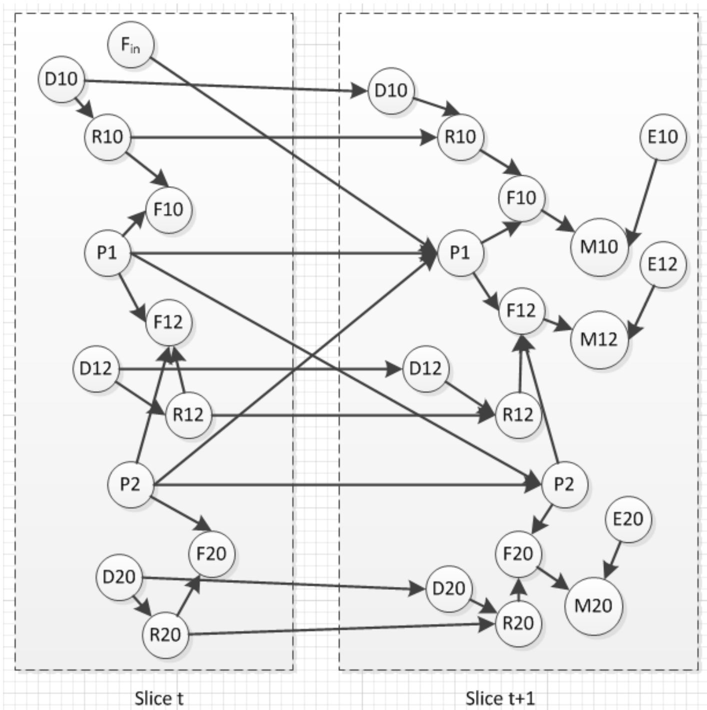


Figure III.1: Complete DBN for Two-tank System

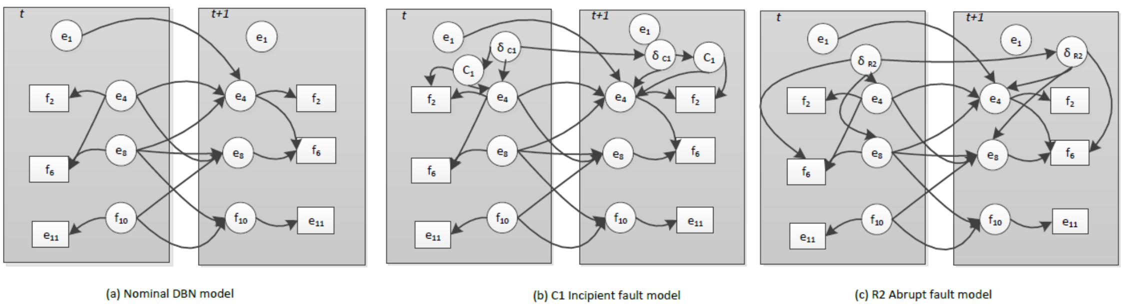


Figure III.2: Dynamic Bayesian Model for Electrical System

represent the temporal relationships between nodes connected.

In addition, Figure III.2(a) shows another example of a third-order electrical system nominal DBN model presented in the previous chapter . It is also represented with state variable $e_4$, $e_8$ and $f_{10}$ drawn as circle, measurements $f_2$, $f_6$, $e_{11}$ drawn as rectangle, and input source variable $e_1$. Inter and intra links are appropriate put according to their relationships (temporal or algebraic) from TCG.

DBNs based diagnostic method requires not only the representation of nominal but also the faulty system behavior. Tracking of faulty behavior requires DBNs to capture three important types of fault effects: burst failures, measurement failures and parameter drift failures. Some failures are persistent that can be in two time slices, like burst failure. Some can be transient failure, which only be in time $t+1$ slice, like the measurement failure. Noted that measurement failure sometimes can be persistent failure as well. Thus, two sets of nodes will be added. The first set corresponds to parameters that represent the fault hypotheses, such as the resistance variable. The second set are discrete-valued nodes that are in 1-1 correspondence with the fault parameters, shown as nodes $D_i$ in the graphical model Figure III.1. They indicate the absence or presence of burst failure or drift failure for that connecting parameter. For such binary fault node, we may denote 0 to be no fault. Besides, additional across time links have to be added as well, such as $D12_t \rightarrow D12_{t+1}$ and $D20_t \rightarrow D20_{t+1}$.

(2) *Quantitative level*: DBNs provide a convenient and compact representation that allows us to model very large and complex systems with a mixture of both discrete and continuous variables. Quantitative level, one needs to designate conditional probability distribution for each variable. Three kinds of information must be specified: the prior distribution over the state variables, $P(X_0)$; the transition model $P(X_{t+1}|X_t)$; and the sensor model $P(Y_t|X_t)$. With the network topology of the connections between successive slices and between the state and evidence variables constructed at the qualitative level, the conditional dependence relationships are clear for each node. Because we assume the stationary process and Markov process, it is most convenient for us to simply specify them for the first slice.

Figure III.1 includes all possible faults in the system. However, the number of possible faults can be really large in complex systems causing complexity issues in tracking diagnostic behavior. For each types of failures in $D$, we need to specify the probability of its presence.

As an example, we might represent a conditional distribution of a continuous node with a discrete parent as a conditional Gaussian. Formally, for a variable $X$ with parent set $D$, we can specify a CPD as follows: for every value $d \in SET[D]$, the CPD has a parameter $\mu_d$ and $\sigma_d^2$; the conditional distribution is then:

$$p(X|d) = N(\mu_d; \sigma_d^2)$$

For the whole DBNs model, given $d_1, d_2, ..., d_t$ as particular instantiation of the discrete variables at time $1, ..., t$. Hence, the current state probability distribution, called as belief state at time $t$ (Koller and Lerner, 2000), the posterior distribution over current state, given all the observations to date, is a multivariate Gaussian over $X_t$.

### III.3 Model Construction

Until now, we have demonstrated many interesting aspects of diagnostic models that can be represented in the DBNs model. (Roychoudhury, 2009) describes a way to construct it from *temporal causal graph* (TCG) framework, which can be viewed as skeleton for appropriate DBNs model. Besides, many types of failure modes can also be incorporated into this nominal model.

A TCG can be described as a diagnosis model that captures dependencies (both algebraic and temporal) between system variables as a causal structure. The TCG can be derived directly from the bond graph model of the physical system. In TCG, node represents effort (pressure) and flow (water flow rate). Two types of arcs existed: one with temporal arcs, annotated with $d_t$ and the other one without temporal arcs, labeled as $\{=,1,-1,R,R^{-1}\}$. Temporal relations in the TCG are associated with the energy storage elements, i.e., the tanks. All other relations in the TCG, e.g., the pressure-flow relations imposed by pipes and the idealized $0-, 1-$ junction relations, are algebraic. (Roychoudhury, 2009)'s method involves three steps: (i) for every effort (or flow) variable associated with a C-element (or I-element) in integral causality, insert a corresponding displacement (or momentum) variable in the system TCG, (ii) "simplify" this TCG so that it contains the state, measured and input variables only, and (iii) construct the system DBN from this simplified TCG.

*Example*: for every node in Figure III.2(a), like input, state, measurement variables, in the simplified TCG, measurement $f_6$ is algebraically related to the state variable $e_4$ and $e_8$. Hence, we draw causal links between them, such as $e_4 \to f_6$, $e_8 \to f_6$. Similarly, we can have other intra-slice links. All other edge labels of the simplified TCG contains $dt$ label indicating an integrating relation, e.g., the edge $e_8 \xrightarrow{dt} e_4$. Hence, we draw such inter-slice causal link $e_8^t \to e_4^{t+1}$ in the DBN.

It is easy to derive DBNs model from TCG, once we place all the state variables and measurement variables in the graph with two replicated time slices. State variables need to project forward, therefore, we have temporal arcs like $X_t^i \to X_{t+1}^i$. If there is a temporal relation from node $X_j$ to node $X_i$ in TCG, we add an arc from $X_j^t \to X_i^{t+1}$. If not, we only draw an arc inside the time slice, like $X_j^t \to X_i^t$ and $X_j^{t+1} \to X_i^{t+1}$. At last see if there is an input left, traverse in TCG from the input variable, have a temporal arc to the first state variable it hits. Besides, we also need to add failure nodes into this model. For persistent faults, such as burst fault and parameter drift failure, we add changing component to be the parent node of its direct effect variables, and add discrete node $D_i$ to be changing component's parent node, indicating the presence of faults. Since they are persistent, they should be replicated to the next time step, and have temporal arcs with each other. Unlike measurement faults, which is only transient, there is only corresponding nodes in time slice $t+1$. We then add $M_i$ to be a normal distribution around measurement variable with small variance when another new node $E_i$ is false (no such fault), but with a much larger variance when $E_i$ is true (fault exists).

*Example*: Figure III.2 (b) and (c) the faulty DBN model of electrical system, the one with incipient fault $C_1^{+i}$ and the one with abrupt fault $R_2^{+a}$. For incipient fault we include an extra stochastic variable $\delta_{c1}$. Assuming that the slope is constant, i.e., slope $\delta_{c1}(t+1) = \delta_{c1}(t)$. The fault parameter $C_1(t)$ is included as an additional stochastic variable that evolves according to the equations $C_1(t+1) = C_1(t) + \delta_{c1}(t)$, and replaces all the occurrences of $C_1$ in the nominal model. Similarly, for abrupt fault $R_2^{+a}$, we also have extra state variable $\delta_{R2}$. Assuming that the magnitude of this bias is constant, i.e., $\delta_{R2}(t+1) = \delta_{R2}(t)$, where $t \geq t_f$. We generate the faulty system model by replacing all occurrence of $R_2$ in the nominal model with $(R_2 + \delta_{R_2}(t))$.

### III.4 Diagnostic Reasoning

Having set up a DBN model with failure modes and making the single fault assumption, we reduce the diagnostic reasoning problem of fault isolation and identification into tracking of conditional dynamic system behavior. The tracking algorithm is quite classic, known as forward propagation, which is to maintain the belief state at time $t$. This recursive process at every time step shown as:

$$P(X_t|e_{1:t}) = P(X_t|e_{1:t-1}, e_t)$$

$$= \alpha P(e_t|X_t) \sum_{x_{t-1}} P(X_t|x_{t-1}) P(x_{t-1}|e_{1:t-1})$$

where $\alpha$ is the normalization factor, second term as sensor model responsible for the correction given evidence $e_t$ and transition model after that is responsible for one-step prediction. Filtered estimate $P(x_{t-1}|e_{1:t-1})$ known as *forward message* could be viewed as a "message" propagated forward along the time sequence.

We use a separate DBN-based observer implemented using a particle filter scheme, PF to estimate the augmented state variables that includes the fault hypothesis. We can instantiate separate PFs since we have the assumption of only single fault occur in the system and these faults can then be independent with each other. Besides, it could allows us to avoid the famous sample impoverishment problem to a good extent using our qualitative TCG based fault isolation method for reducing the number of potential fault hypotheses. Different from previous work (Roychoudhury, 2009), this thesis also extends it by developing an auxiliary particle filter approach (Pitt and Shephard, 1999) to track system behavior and estimate dynamic system state from noisy measurements. The previous approach suffered from long convergence time and in case of some faults low accuracies in estimating the fault parameter values. In this thesis we will also do a comparison between these two following PFs on the performance.

Particle filtering algorithms solve the fundamental problem of recursive Bayesian filtering, via a discrete approximation to the filtering density. We now discuss a couple of prominent particle filtering algorithms, GPF and APF.

#### III.4.1 Generic Particle Filter

Particle filters represent a class of sequential importance sampling algorithms that are commonly used for tracking and estimating the true system behavior using DBNs (Koller and Lerner, 2000) . The key idea is to represent the required posterior density function by a set of random particles with associated weights. As the number of samples becomes very large, this Monte Carlo characterization becomes an equivalent representation to the usual functional description of posterior probability density function, and this sequential importance sampling filter approaches the optimal Bayesian estimate.

For the tracking problem, we always have a target, which is the state vector, $x_t$ at each time step $t$. If we denote the measurement as $z_t$, the goal is to derive a sequence of particles $\{x^i_{0:k},\ \omega^i_k\}$, where $\{x^i_{0:k},\ i = 0, \ldots, N_s\}$ is a set of support points with associated weights $\{\omega^i_k,\ i = 1, \ldots, N_s\}$, such that the posterior distribution $p(x_t \mid z_t)$ is optimized. Because this is a probability distribution, therefore the weights must satisfy

$$\sum_i^N \omega_i = 1, \tag{III.1}$$

where we have a sample size of N = 200. Consequently, the best approximation for the posterior at each time step is the mean state of all the particles:

$$E(x) = \sum_i^N \omega_i x_i \tag{III.2}$$

We choose the importance density to be the prior:

$$q(x_k \mid x^i_{k-1}, z_k) = p(x_k \mid x^i_{k-1}) \tag{III.3}$$

A generic particle filter algorithm is summarized in Algorithm 1. The only difference from bootstrap filter is that it only resamples when samples indicate severe degeneracy. The effective sample size $N_{eff}$ is defined as:

$$N_{eff} \quad = \quad \frac{1}{\sum_{i=1}^{N_s} (\omega^i_k)^2} \tag{III.4}$$

Whenever the effective size of samples falls off the threshold value, which means a significant degeneracy is observed, we need to do resampling step. The basic idea is to eliminate particles that have small weights and to concentrate on particles with large weights, which involves generating a new set of samples $\{x^i_k\}_{i=1}^{N_s}$ by resampling $N_s$ times from equation III.3 and the weights are reset to $\omega^i_k = 1/N_s$. In this thesis, we implement the systematic resampling(Arulampalam et al., 2002), which only takes $O(N_s)$ time and minimizes the MC variation.

| Algorithm 1 Generic Particle Filter on DBNs |
|---|
| **Input:** samples and associated weights at time step k-1; a DBN D={X,Z,U,Y} |
| **FOR** each particle i, from 1 to $N_s$: |
| – Draw sample $x_k^i \sim$ q($x_k$ \| $x_{k-1}^i$, $z_k$) |
| – Assign weight to each sample, $\omega_k^i$ = $\omega_{k-1}^i$ p($z_k$ \| $x_k^i$) |
| **END FOR** Calculate total weight |
| Normalize all the weights according to total weight |
| Calculate $N_{eff}$ |
| **IF** $N_{eff} < N_T$ |
| – Resampling |
| **END IF** |

Table III.1: Generic Particle Filter.

### III.4.2 Auxiliary Particle Filter

The Auxiliary Particle Filter (APF) is introduced by(Arulampalam et al., 2002) as a variant of the generic particle filter. This filter obtains a sample from the joint density $p(x_k,\ i \mid z_{1:k})$ and then omits the indices $i$ to produce sample $x_k$ we need from pdf $p(x_k \mid z_{1:k})$. The index $i$ is called the auxiliary variable. The importance function chosen to draw the sample is defined as:

$$q(x_k, i \mid z_{1:k}) = p(z_k \mid \mu_k^i)p(x_k \mid x_{k-1}^i)\omega_{k-1}^i, \tag{III.5}$$

where $\mu_k^i$ is some characterization of $x_k$, given $x_{k-1}^i$, and the assigned weight is proportional to the right hand side of equation below:

$$\omega_k^i = \frac{p(z_k \mid x_k^i)}{p(z_k \mid \mu_k^i)} \tag{III.6}$$

The algorithm for the auxiliary PF is summarized as Algorithm 2. Note that we use the same resampling method as GPF.

Both algorithm 1 and 2 use the same resampling scheme, and only when the algorithm detects severe weight degeneracy it resamples. However, for GPF the old weights at time $t-1$ will propagate to time $t$ which tends to cause weight degeneracy easily. APF in Algorithm 2 solves this problem by selecting the proposal density function shown as equation III.5. Based on this new proposal function, the new weight $\omega_k^i$ does not need to depend on old weight $\omega_{k-1}^i$.

| Algorithm 2 Auxiliary Particle Filter on DBNs |
|---|
| **Input:** samples and associated weights at time step k-1; a DBN D={X,Z,U,Y}<br>**FOR** each particle i, from 1 to $N_s$:<br>– Draw index $i$ from $q(i \mid z_{1:k}) \sim p(z_k \mid \mu_k^i)\ \omega_{k-1}^i$<br>– Draw sample $x_k^i$ from $q(x_k \mid i, z_{1:k}) \sim p(x_k \mid x_{k-1}^i)$<br>– Assign weight to each sample, $\omega_k^i$ according to equation III.6<br>**END FOR** Calculate total weight<br>Normalize all the weights according to total weight<br>Calculate $N_{eff}$<br>**IF** $N_{eff} < N_T$<br>– Resampling<br>**END IF** |

Table III.2: Auxiliary Particle Filter.

### III.5 Summary

In this chapter, we describe a general approach called dynamic Bayesian network (DBN) for diagnosis of complex nonlinear systems. Dynamic Bayesian Networks is a name of a model that describes a system that is dynamically changing or evolving over time. The model enable users to monitor and update the system as time proceeds, and even predict further behavior of the system. It allows us to represent very complex stochastic systems, including ones that involve both discrete and continuous variables. Particle filtering provides a general-purpose inference algorithm that can be applied to virtually any DBNs. Thus, it allows us to deal with extremely rich class of dynamic systems.

DBNs are constructed under Markov assumption, where each state $X_t$ at time $t$ only depends on previous state $X_{t-1}$ at time $t-1$. The models at each slice are based on Bayesian network. Arcs connecting two nodes together denotes direct dependencies. Nodes can be discrete or continuous variables. At the initial state, to finish constructing the model, it needs to specify conditional probability table to each node. The data could be from knowledge-based. During model construction, the network structure of DBNs could be based on domain independent bond graph modeling language, which leads to an accurate result.

Qualitatively, DBNs have across time links between two slices denoting state transitions, where the next time slice is replicated from the previous slice. With such property, DBNs are better to be used in monitoring system behaviors. Quantitatively, DBNs can have both continuous and discrete nodes. Multivariate Gaussian distribution can be derived from statistical observation data given by expert.To maximize the joint possibility of true fault hypothesis, DBNs are used in diagnosis domain. However, if the number of fault hypothesis is large, exact inference could be computationally intractable. Approximate inference methods like particle filter are widely used.

The DBNs method could be used in complex physical system domain, like the electrical system, two-tank system, and also medical diagnosis domain with decision support. For a system with conditional linear Gaussian distribution, a distribution with a multivariate Gaussian component for each instantiation of the discrete variables, even simple as every continuous variable has at most one binary discrete ancestor, the inference is NP-hard in complexity. There doesn't exist a polynomial time approximate inference algorithm.

In all, a general probabilistic model, like DBNs, it has several advantage and disadvantages. Pros: (i) since it is a complete model of the system, it includes within it the likelihood of different types of failures, as well as a distribution over the relevant system parameters. Many challenging problem such as ranking possible failures, handling of multiple simultaneous failures and robustness to parameter drift can be solved within a probabilistic tracking framework; (ii) it is flexible which can allow arbitrary probability distribution and nonlinear phenomena; (iii) each node represents a specific concept; (iv) it can handle large number of variables. Cons: (i) since the number of failure modes of such complex system grows exponentially, inference of DBNs is generally intractable, even for some approximate methods; (ii) if approximate inference using sampling algorithm, the speed is quite slow making them unsuitable for large models; (iii) it requires specialized statistical knowledge to ensure convergence of the dependent samples to a reliable result.

# CHAPTER IV

## Case Study - The Reverse Osmosis System

We demonstrate our DBN-based diagnosis scheme by applying it to the Reverse Osmosis (RO) system, a subsystem of the Advanced Water Recovery System (AWRS) used to reclaim waste water generated on a long term space mission. A real AWRS testbed was designed and built for long-duration manned missions (Kortenkamp and Bell, 2003) and (Pickering et al., 2001). A previous project in Institute of Software Integrated Systems (ISIS), Vanderbilt University, has applied TRANSCEND diagnosis scheme to the RO system. However, qualitative scheme lacks complete diagnosability primarily because of the ambiguities introduced by the qualitative reasoning scheme. In this thesis, we employ a combined qualitative-quantitative scheme combining the TRANSCEND approach with a particle filter based implementation of a DBN-based scheme to detect, isolate, and estimate the fault magnitude after faults occur in the system. There are two main tasks: apply DBN model based diagnosis approach to RO system, and compare the performance of generic particle filter with auxiliary particle filter using as inference scheme for the DBN model.

### IV.1 System Description

Figure IV.1 shows the schematic of the RO system (Szarka, 2011). The RO system gets wastewater from Biological Water Processor (BWP) subsystem, which removes inorganic matter and particles from the water. And after a post processing step, where Ultra-Violet treatment is applied to the output of the RO to remove trace contaminants and generate potable water. This process typically cleans 85% of the water. Based on the valve, which controls the direction of the liquid flow in the back-flow pipe, the system could operate in three modes. In the *primary* mode (valve setting 1), the feed pump keeps pushing water extracted from BWP into the main RO loop. The recirculation pump boosts the liquid pressure as it flows into the membrane module. The flow through membrane module causes dirt to accumulate in the membrane, which increases the resistance to flow through it, thus causing the outflow from the system to decrease with time. Apart from water through membrane, rest of the water flows back to the main loop. The tubular reservoir helps balance fluctuations in the flow through the loop. After a user-defined time interval, the RO system will transition to *secondary* (valve setting 2) mode in which the liquid flow in the back-flow pipe and *purge* (valve setting 3) mode in which the recirculation pump is off and the liquid is pushed through the drain into AES. In this thesis, we only consider RO system operating in the primary mode, where the system is a continuous dynamic system, and perform model-based diagnosis without hybrid discrete mode transitions.

RO system is modeled in GME using the BDM paradigm. Via the model interpreter, buildscript and

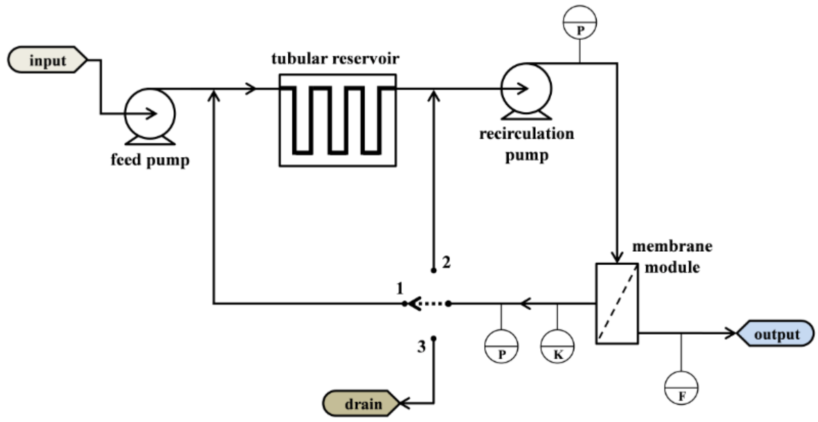


Figure IV.1: Schematic of the RO system

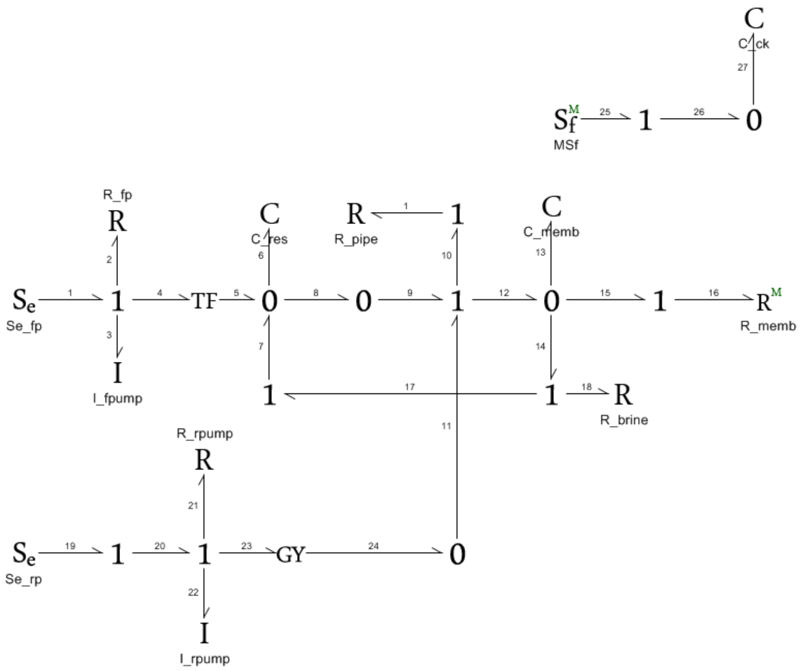


Figure IV.2: Bond graph Model of RO system in primary mode

the simulation operation can be performed on the constructed model in Matlab. Figure IV.2 shows the bond graph model of RO system in primary mode with bond number specified. Parameter values for the model are extracted from a previous paper (Biswas et al., 2004). Table IV.1 lists all the nominal parameter values. The appropriate state-space equation forms describing the state variables and measured value relations were then derived from the bond graph, shown as equation IV.1. There are four states and four measurements considered, detailed in the next section. We also build a simulink model based on such equations to validate the model.

$$
\begin{aligned}
\frac{d_{f_3}}{d_t} &= e_1 - f_3 R_{fp} - m_1 e_6 \\
\frac{d_{f_{22}}}{d_t} &= e_{19} - f_{22} R_{rp} - \frac{m_2}{R_{pipe}}(m_2 f_{22} + e_6 - e_{13}) \\
\frac{d_{e_6}}{d_t} &= \frac{1}{C_{res}}(m_1 f_3 - \frac{e_{13} - e_6}{R_{brine}} - \frac{1}{R_{pipe}}(m_2 f_{22} + e_6 - e_{13})) \\
\frac{d_{e_{13}}}{d_t} &= \frac{1}{C_{memb}}(\frac{1}{R_{pipe}}(m_2 f_{22} + e_6 - e_{13}) - \frac{e_{13} - e_6}{R_{brine}} - \frac{e_{13}}{MR_{memb}})
\end{aligned}
\tag{IV.1}
$$

In the bond graph model, two physical domains involved. (1) mechanical domain, two pumps are modeled as the sources of effort ($S_{efp}$ and $S_{erp}$), which maintain a constant torque of the rotor. The rotational inertia of the rotor and the power dissipation associated with friction are modeled with inertias and resistors connected to the sources with 1-junctions ($I_{fp}$, $I_{rp}$, $R_{fp}$, $R_{rp}$). (2) hydraulic domain, the tubular reservoir and the membrane module are modeled as a capacitor ($C_{res}$) and a capacitor ($C_{memb}$), respectively and there is also a modulated resistor ($MR_{memb}$). The value of $MR_{memb}$ is affected by the conductivity value $K$, the measure of the concentration of impurities in the water.

$$MR_{memb} = 0.202 * (((K - 12000)/165 * 4.137e + 011) + 29 * 4.137e + 011)$$

The conductivity value is some function of signals coming from the hydraulic domain. Because the current conductivity value affects the resistance of the membrane, we include a bond graph fragment at the right-top of Figure IV.2 to compute the conductivity value at each time step in the system. It is an imaginary representation, associated with the effort of the imaginary capacitor element $C_{ck}$. In the primary mode, there is one modulated sources of flow, flow rates of the back-flow pipe, that is used in the bond graph to compute the conductivity value $e_{27}$.

$$MS_f = (memb_{inflow} * ((backflow/1.667e - 08 * 6) + 0.1)/1.667e - 008),$$

where the $memb_{inflow}$ is the flow rate of $R_{pipe}$, i.e., $f_{10}$, and $backflow$ is the flow rate of back-flow pipe $R_{brine}$, i.e. $f_{18}$. This is also one reason that make the system nonlinear. The pipe between the membrane module and the reservoir is modeled with one resistor ($R_{pipe}$) in the primary mode. The back-flow pipe is modeled with one resistance ($R_{brine}$) when the water circulates.

| **Param.** | **Unit(SI)** | **Value** |
|---|---|---|
| $C_k$ | ml/min · mS | 565 |
| $C_c$ | $m^5$/N | 1.5 |
| $C_{brine}$ | $m^5$/N | 8 |
| $R_{brine}$ | N·/$m^5$ | 220 |
| $C_{memb}$ | $m^5$/N | 0.6 |
| $R_{memb}$ | N·/$m^5$ | 26.0 |
| $I_{ep}$ | N·m·$s^2$ | 2 |
| $R_{ep}$ | N·s·m | 0.1 |
| $I_{fp}$ | N·m·$s^2$ | 0.1 |
| $R_{fp}$ | N· $m^5$ | 0.1 |
| $R_{pipe}$ | N· $m^5$ | 69.0 |

Table IV.1: Nominal Values for the RO system Bond Graph Parameters.

### IV.2 Experiment Setup and Result Analysis

There are two main tasks. First, we will demonstrate the DBN model based diagnosis approach by applying it into RO system Figure IV.2. Second, compare the performance of generic particle filter with auxiliary particle filter using as inference scheme for the DBN model. All the components in the bond graph are parameters that could change value when faults occur. In this subsystem, we have four state variables: $f_3$, $f_{22}$, $e_6$ and $e_{13}$, which are flow rate through feed pump, recirculation pump, and pressure of the recirculation pump, membrane, respectively. Four measurements have been used: (i) the pressure of the permeate at the membrane, $e_{14}$, (ii) the flow of the effluent, $f_{16}$, (iii) the pressure of the liquid in the return path of the recirculation loop, $e_{18}$, (iv) the flow of back-flow loop for changing conductivity value studies, $f_{14}$. Figure IV.3 shows the data collected from simulink model of the nominal RO system. From top to bottom, shows the membrane pressure ($P_{memb}$), the water conductivity value ($K$) and outflow of membrane module ($F_{pem}$).

As described before, we only consider the system running on primary mode without any hybrid mode transition, where the input flow is mixed with the water in the primary recirculation loop. The recirculation pump boosts the liquid into the membrane. Although empirical information on the noise is not available, process and measurement noise were simulated as zero mean white Gaussian noise with variances at 3% and 2%, respectively. Figure IV.4 shows the filtered graph for nominal system without any fault. The red circle is the actual measurement and the blue dash-dot line is the system behavior after filtered. Fault scenarios were created that correspond to incipient fault in the connecting pipe $R_{pipe}$ and abrupt fault in the membrane $R_{memb}$.

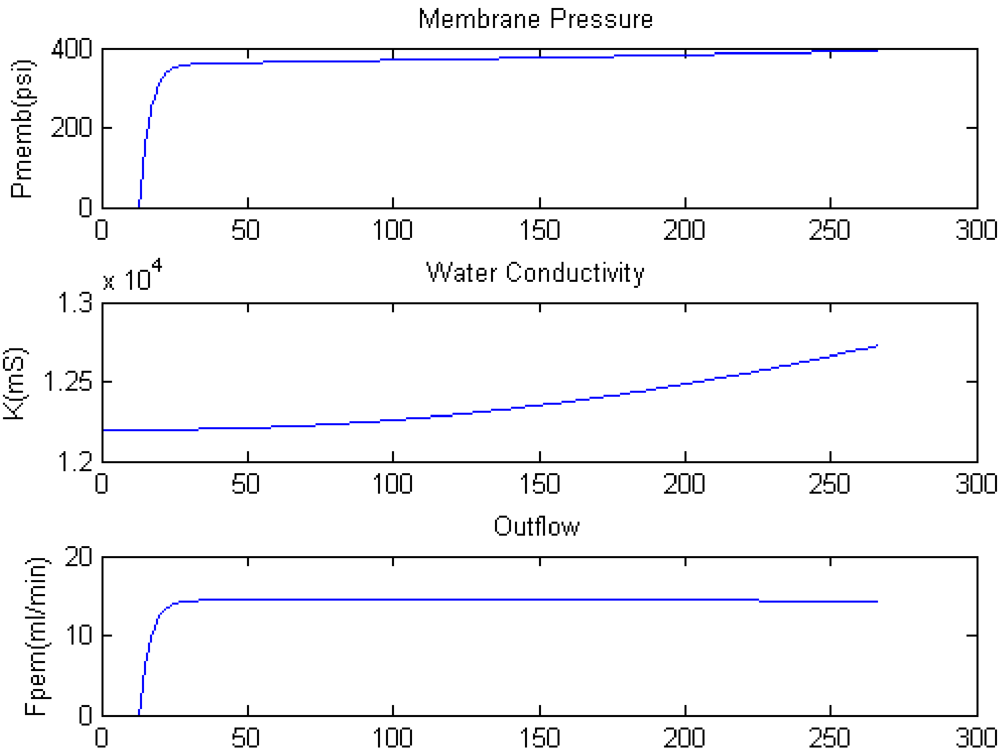


Figure IV.3: RO system Nominal Behaviors Without Noise

### IV.2.1 Diagnosis Experiment

During our experiments, we apply the DBN model based fault diagnosis scheme in different fault scenarios. The system runs for a total 300 time steps by simulation using the Simulink/MATLAB environment, with fault introduced at time step t=165 time point. The nominal data from Simulink model generated by BDM model were then saved into a mat file, and used to build our DBN-based nominal observer for the system. The physical system being monitored and the nominal observer receive the same input signals, and the system output, i.e., the actual system measurements from BDM model, are labeled as $y[k]$, and the observer estimates are labeled as $\hat{y}[k]$. The system residual vector is then computed as $y[k]-\hat{y}[k]=r[k]$ at time step $k$. The fault detection scheme uses hypothesis testing methods to determine if the computed residual signals imply a fault in the system. It has to be robust to measurement and process noise. The output of this detection component is a vector of binary variables, $b$, representing the fault signature for the system, and a set of parameters, $\theta$, that describe the change in the residual signal. It is given to the next fault isolation and fault parameter identification processing units.

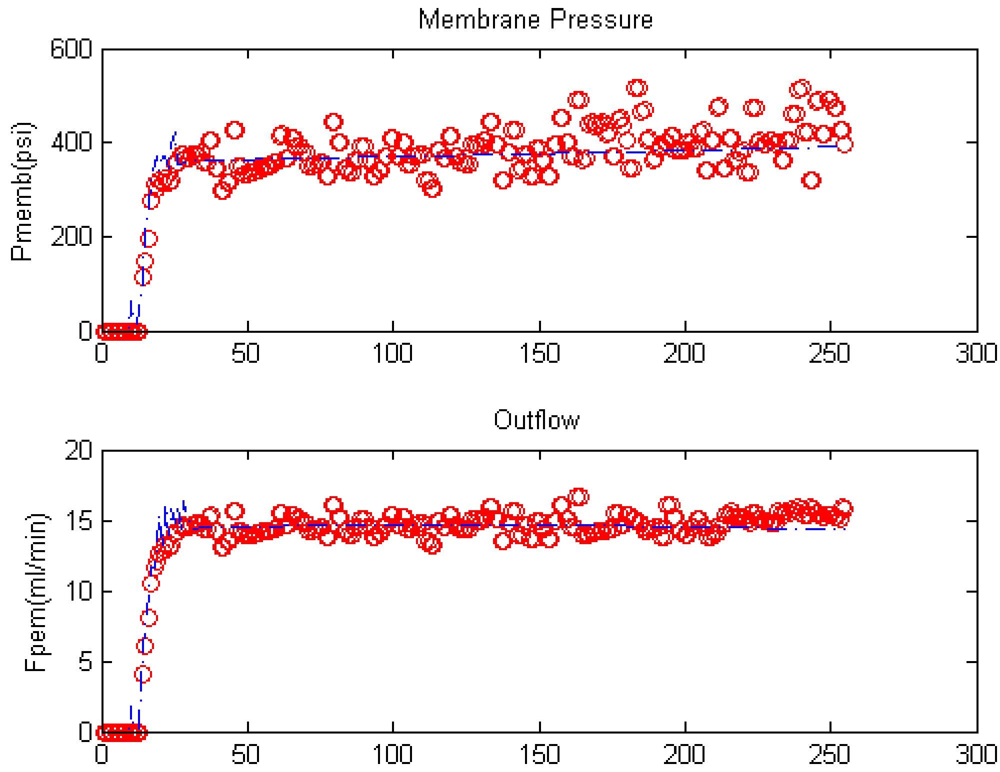


Figure IV.4: Nominal System Behavior in primary mode filtering using PF

Followed by detection part, fault isolator aggregates all the data from output of detection components to build a fault signature vector based on the value of $b$ and the direction of the change from $\theta$. The fault signature is then looked up in a table of possible faults to find the fault candidates which identify the fault and faulty component. Fault isolator then sends the possible fault candidates, and all of the residuals and raw sensor data to the fault identifier. The fault identifier narrows down the set of fault candidates set to the most likely candidate. Once this qualitative scheme refines the number of fault hypotheses to a pre-defined small number, or $s$ timesteps have elapsed, we start our quantitative scheme. It performs both fault isolation and identification. A bunch of faulty DBN models corresponding to each remaining fault hypothesis in the set is initiated. DBN models for RO system with abrupt fault and incipient fault are quite similar to Figure III.2 using the same construction process as we described in chapter III. See Figure IV.5 of $R_{pipe}^{+i}$ faulty DBN model.

We then run a particle filter for each of these DBN fault models, taking the measurements from the time of fault detection point, $t_d$, as input. As more observations are obtained, only the PF that uses the correct

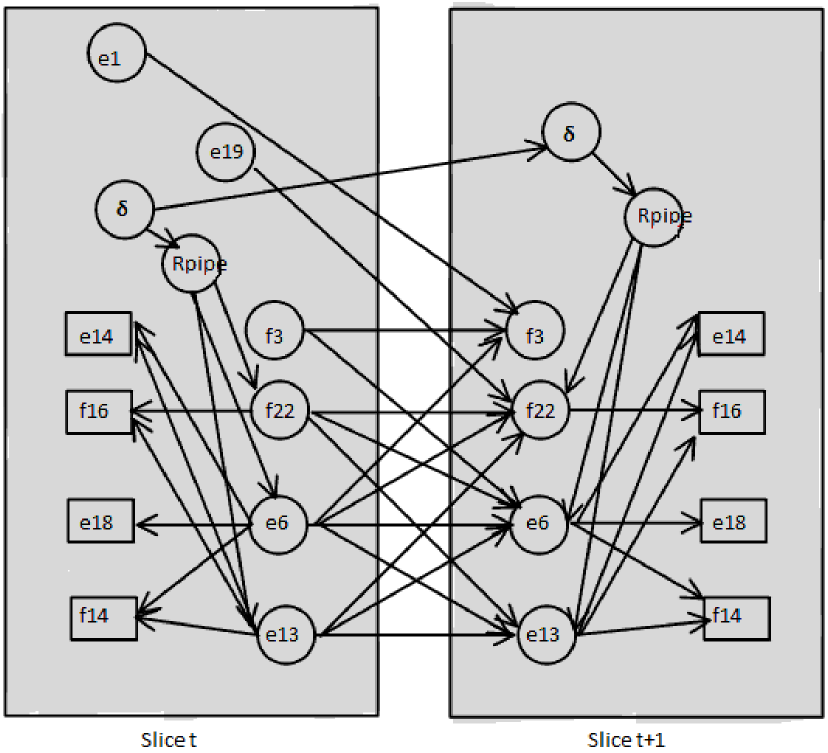


Figure IV.5: $R_{pipe}^{+i}$ faulty DBN model

faulty DBN model, should be converging to the observed measurements, while the observations estimated by the PFs that use the incorrect faulty DBN mdoels should gradually deviate from the actual observed faulty measurements. See from the Figure IV.6. The fault $R_{pipe}^{+i}$ is introduced at $t = 165$ and the fault detector signals this deviation at time step around $t = 167$. The PFs on these two faulty DBN models, taking as inputs, only the measurements at time points $t > 167s$, the time of detection of the fault. Eventually, the statistical test indicate that the observations estimated by the PF applied to hypothesis of $R_{fp}^{+i}$ has significantly deviated from the observed faulty measurements, correctly isolating fault $R_{pipe}^{+i}$ as our true fault at this scenario. The estimated measurements, $P_{memb}$ and $F_{pem}$ from these two fault models are shown in Figure IV.6. The red cross markers are the actual fault measurements and the blue line is consistent with the actual measurements, which uses faulty DBN model of $R_{pipe}^{+i}$. The green square line shows a great deviation from the actual measurement, which uses faulty DBN model of $R_{fp}^{+i}$ and the fault will then be dropped. Around $t = 185$, the blue dash-dot line converges to the actual measurements. and around t=190, the green line significantly deviate from the actual measurements.

FigureIV.7 shows the incipient fault $R_{pipe}$ parameter estimation. Originally $R_{pipe}$ is running on nominal

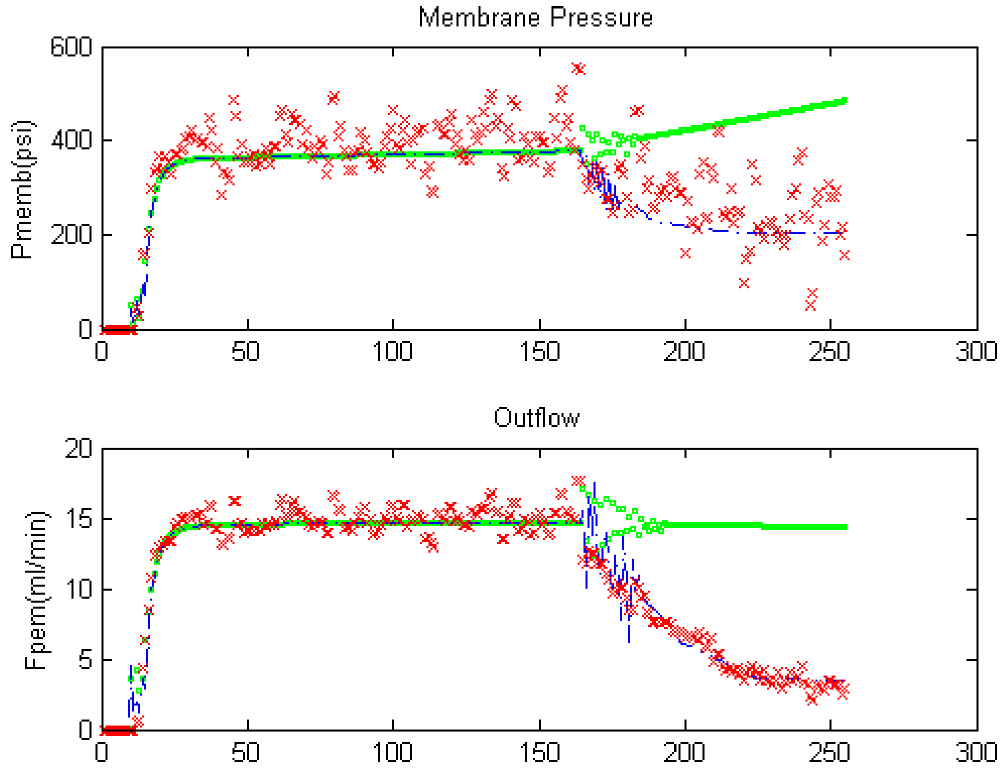


Figure IV.6: Estimated observations for fault mode $R^{+i}_{pipe}$ with two faulty DBN models

value 69. At the time of fault detection, we replace all the occurrence of $R_{pipe}$ with a linear change drift term $\delta_{Rpipe}$, implemented with a constant slope $S_{estimate}$. Around $t = 185$, the particle filter converges to the true faulty measurement data. Therefore, the fault parameter $R_{pipe}$ is also estimated correctly. And the slope is estimated to be 0.47, while the true injected fault slope is 0.5. This experiment demonstrate that our DBN approach can maintain the current belief state and propagate to the next time step for a uncertain dynamic system.

### IV.2.2 PFs Comparison Experiment

Now that we have demonstrated our DBN model based diagnosis scheme using RO system as a case study. We turn to investigate the different performance using different inference scheme for DBN models. We use time to convergence (CT) and mean-square error (MSE) as metrics to compare the performance of the two particle filters: GPF and APF. For every fault scenario, we run the experiment 10 times and then calculate the mean value and standard deviation (SD) of the performance parameters, as shown in Table IV.2. DBNs used

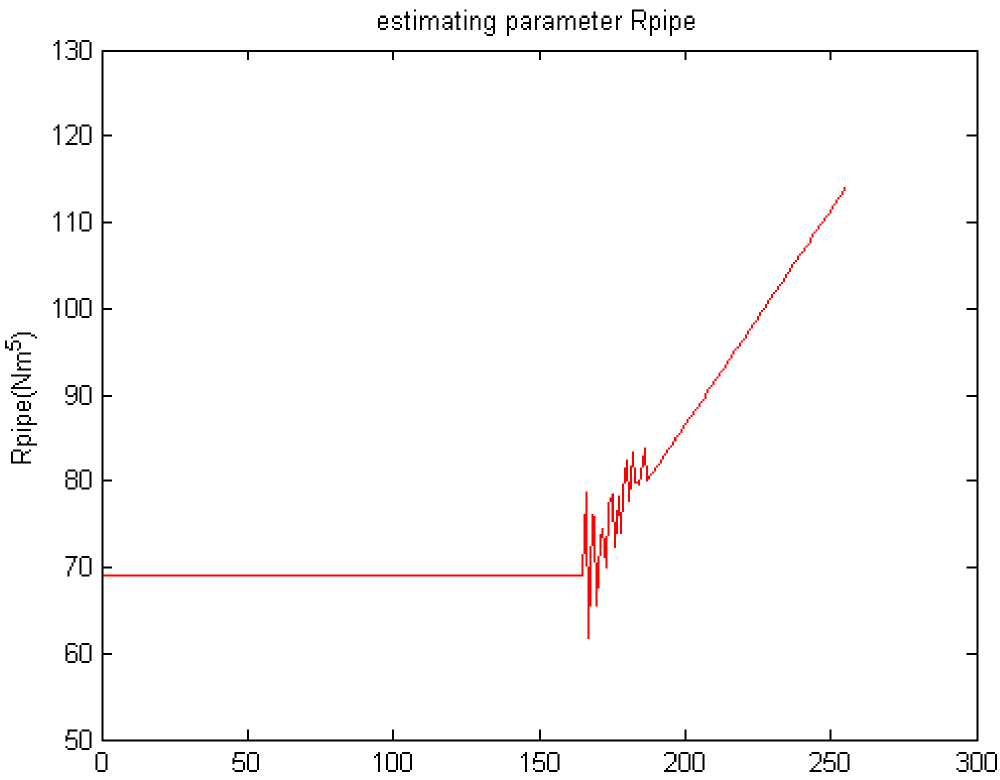


Figure IV.7: Fault Parameter Estimation

in all the experiments have the same configuration which allows a better comparison of their performance. Each PF experiment uses 200 particles at each time step. All the experiments were run on the same desktop, assuming the same workload. The desktop used for this experiment is an Intel Core i5 at 2.4 GHz. It has 4GB of RAM memory and runs Mac OS X 10.7.2 operating system.

In order to provide a reproducible and algorithm-independent assessment of the tracking ability of a particle filter applied to the fault diagnosis problem of nonlinear electrical system, we choose the following performance parameters as our metrics.

**RT**: run time, measures the overall runtime efficiency. RT reports the run time for 300 time steps.

**MSE**: mean-square error, for each state of the system, the particle filter delivers an estimate of the current state as mean of the particles, $\hat{x} = E(x_t)$. The square error $\varepsilon_t$ for time point $t$ is computed as $\varepsilon_t = (\hat{x} - x_{true})^2$. The MSE value then corresponds to the variable $\varepsilon_t$ averaged over the total number of time units in processing the system states.

**CT**: time to convergence. To measure CT, we define a threshold value $\varepsilon_t$. For time unit $t$, the particle

Table IV.2: GPF and APF in four scenarios

| Faults | GPF | | | APF | | |
|---|---|---|---|---|---|---|
| | RT(SD) | MSE(SD) | CT(SD) | RT(SD) | MSE(SD) | CT(SD) |
| nominal | 16.25(0.52) | 0.51(0.08) | 183(10) | 24.25(0.48) | 0.20(0.04) | 176(12) |
| $R_{pipe}$ incipient | 17.04(0.55) | 0.59(0.12) | 194(11) | 28.77(0.45) | 0.33(0.09) | 185(12) |
| $R_{pipe}$ abrupt | 17.36(0.49) | 0.66(0.11) | 196(11) | 29.54(0.49) | 0.36(0.08) | 188(10) |
| $R_{memb}$ incipient | 17.19(0.53) | 0.71(0.13) | 190(15) | 29.63(0.49) | 0.40(0.08) | 183(14) |
| $R_{memb}$ abrupt | 16.98(0.45) | 0.72(0.08) | 192(14) | 29.35(0.54) | 0.39(0.09) | 185(12) |

filter is said to be converging toward the true state $x(k)$ if the latter lies within one standard deviation $\varepsilon_t$ from the estimated state $\hat{x}$. In other words, the particle filter is convergent if the following inequality holds: $MSE \leq \varepsilon_t$.

Table IV.3: GPF and APF for parameter estimation

| TrueParameter | GPF | | APF | |
|---|---|---|---|---|
| | Estimate | PercentageError(%) | Estimate | PercentageError(%) |
| $R^{+a}_{pipe}(100)$ | 91.88 | 8.12 | 96.36 | 3.64 |
| $R^{+a}_{memb}(50)$ | 45.73 | 8.54 | 47.92 | 4.16 |

Table IV.2 shows the particle filter is efficient and accurate for tracking and estimating nominal and fault system behaviors. GPF runs much faster than APF but it needs more time to converge than APF in all the four different scenarios. See Table IV.3, according to the estimation of abrupt fault parameter, APF shows a better accuracy than GPF. Mostly it is due to the reason that GPF uses prior density as the proposal function to produce a posterior which is not so reliable and the weights are very unevenly distributed. On the contrary, APF introduce an auxiliary variable and sample from a joint density which is more close to the empirical filtering density, but it may easily cause sample degeneracy and resamples a lot. Hence, GPF is faster but less accurate.

After it converges we extract the value of this fault hypothesis. For example, $R^{+a}_{pipe}$ was set to have an abrupt value change to 100, and when our fault model is considered to converge to the actual measurement deviation, the value of $R^{+a}_{pipe}$ was estimated as 91.88 using GPF and 96.36 using APF. To be better visualized, we could see from its percentage error and APF got a less error percentage.

Furthermore, from Table IV.4, we can see how GPF and APF works under both low and high levels of noise, where we adjust both process and measurement noise at the same time in this thesis. Compared with the generic particle filter, the auxiliary particle filter works better when noisy level is light. Basically, because it generates points from the samples at $k-1$, which are most likely to be close to the true state. If the process noise is small, so that $p(x_k \mid x_{k-1}^i)$ is well characterized by $\mu_k^i$, which is estimated by the prior, then the APF is often not so sensitive to outliers as GPF, and the weights $\omega_k^i$ are more even. However, if the process noise is quite large, a single point cannot characterize $p(x_k \mid x_{k-1}^i)$ well enough. Hence, it will cost more time to converge than GPF and it will get less accuracy than usual. Also see (Pitt and Shephard, 1999).

Table IV.4: GPF and APF in two types of noise levels

| | GPF | | | APF | | |
|---|---|---|---|---|---|---|
| $R_{pipe}^{+a}$ | RT(SD) | MSE(SD) | CT(SD) | RT(SD) | MSE(SD) | CT(SD) |
| high-level | 18.74(0.55) | 0.85(0.24) | 205(14) | 34.23(0.81) | 0.65(0.21) | 218(16) |
| low-level | 17.36(0.49) | 0.66(0.11) | 193(11) | 29.54(0.49) | 0.36(0.08) | 188(10) |

## IV.3 Summary

In this chapter, we demonstrate the effectiveness and correctness of our DBN model based diagnosis scheme by running diagnosis experiments and PFs comparison experiments on a number of fault scenarios. Reverse Osmosis (RO) system in the primary mode operation designed and built in BDM paradigm was used as our testbed. Followed by the modeling chain described in chapter III, its state-space equation form, bond graph model, TCG and DBNs are created, from which we could understand the system in a deeper level. And with the DBN model, nominal and faulty, particle filters are used to do the diagnostic reasoning and inference. Eventually, only one particle filter that uses the correct faulty DBN model converge to the actual faulty measurement.

Another important contribution of this work is to compare two kinds of particle filters, based on their performance on a number of fault scenarios and different levels of noise. Generic particle filter is faster at the running time. However, from view of time to convergence and mean-square error, auxiliary particle filter shows an improvement with accuracy and efficiency. When the system noisy level is quite high, APF may need more time to converge to the actual value or even lose the ability to converge.

The significant advantage of the DBN model-based diagnosis is that it captures the temporal and causal relations between system variables and component parameters, allowing for very efficient qualitative models and quantitative reasoning methods. This helps overcome some of the limitations that have been observed for analysis of faults in systems with complex nonlinear behaviors. As the experimental results in the previous work demonstrate, the qualitative scheme is always ambiguous, but once the fault set is reduced to a small size by qualitative model, the quantitative estimation techniques can be applied to uniquely isolate the fault and compute the magnitude of change.

In conclusion, we have developed a new DBN model based fault diagnosis method combining the qualitative framework and quantitative scheme for the complex nonlinear dynamic system, the Reverse Osmosis (RO) system, part of water recovery system. The experiment results matched the expected values and further development of this technology will provide the proof of concept that advanced control techniques can form the backbone for autonomy in future long-duration missions (Biswas et al., 2004).

CHAPTER V

# Discussion and Conclusions

## V.1 Summary

The diagnosis problem is designed to determine the current state of a system (nominal or faulty) given a stream of observations from that system (Dearden and Clancy, 2002). Therefore, system state tracking and estimation play a very important role in the online diagnosis framework. However, with nonlinear and non-Gaussian behaviors, it is a challenging task, even with low levels of process and measurement noise. Sequential Monte Carlo methods provide a number of advantages, and PF approaches have been used extensively for system monitoring and diagnosis of hybrid systems (Lerner et al., 2000). But they require many computational resources to get a good approximation of the true belief state. (Narasimhan et al., 2004) use Livingstone 3 to generate a set of candidates and track the stochastic system behavior by look-ahead Rao-Blackwellized Particle Filter scheme. The proposed approach combines QFS to generate fault hypothesis and tracks observed measurements using a PF separately that runs on each faulty DBN model till the particles eventually converge to one of the fault modes. Besides, the auxiliary particle filter has been shown to be a stable, efficient and accurate method for tracking and estimating fault parameters in complex physical systems.

In this thesis, we introduce particle filter reasoning method in our DBN-model based diagnosis framework combining with previous qualitative TRANSCEND scheme. Generic particle filter and auxiliary particle filter are employed and compared based on their performance running on the Reverse Osmosis (RO) system. Carrying out state estimation and behavior tracking in such practical hybrid system is quite a complicated task. That's the reason this thesis only considers the system running on one primary mode, making hybrid system to be a continuous nonlinear system, focusing on diagnosis scheme not on hybrid mode transition. Besides, we have different levels of disturbance and measurement noise. Even low levels of noise can rapidly become detrimental to traditional diagnosis method. Under such adverse conditions, incorporating these observations in the DBN model based diagnosis framework proves to be a substantial advantage. Using data samples generated from BDM models(Szarka, 2011) which correctly simulated the real system, we have furthermore demonstrated that particle filter algorithms based on updated measurements show a high degree of robustness against process and measurement noise. Most of previous work(Arulampalam et al., 2002) prove that for a variety of real scenarios, if the assumptions of the Kalman filter cannot hold and the system is nonlinear, approximate techniques must be employed. Particle filtering approximates the density directly as a finite number of samples. A number of different types of particle filter have been developed, such as SIR, ASIR and RPF, and some have been shown to outperform others when used for particle filter for a particular applications. However, when designing a particle filter for a DBN model, it is the choice of importance density that is critical.

We have developed a better understanding of dynamic Bayesian networks and show that this DBN model can be useful for practical applications, in particular in order to perform fault diagnosis on a complex real world nonlinear dynamic system. We first present an overview of DBN model based method, its representation, its construction method and its reasoning algorithms. As stated before, with the system state space getting larger, it could be computational intractable. The task to find the likely states of the system given sensor readings is NP-hard. This is then we combined qualitative scheme to reduce the number of fault hypothesis to a small number so as to decrease the computation complexity in the diagnosis framework.

### V.2 Future Directions

It is our hope that this thesis demonstrates the usefulness of dynamic Bayesian networks and provides useful algorithms for inference in these models. Obviously, it is quite difficult to answer all the questions that come up in these models, and there is still room for much work to be done. In this section, we will briefly review some exciting research directions in this field.

(i) Diagnosis on Hybrid System with discrete mode transition. Such systems will require new variants of many of the techniques we currently employ in model-based diagnosis including exploiting problem decomposition, compact representations of state spaces, abstractions of problems, and approximation of inference. It is still quite a challenging problem.

(ii) Inference Issues. The heart of the algorithm used for tracking the RO system is the particle filter approximation from chapter III. Although these algorithms were proven to be capable of reliably dealing with a complex real-world dynamic system, it still can be improved in a few ways. We could improve the efficiency of our diagnosis approach by deriving reduced DBN models and running PFs on these reduced-order DBN models instead of on the entire system DBN model.

(iii) Modeling other types of faults. In the case study, we have concentrated on diagnosing abrupt changes and gradually drifts. In physical systems, there is another important type of change, called intermittent fault. It is a malfunction of a device that occurs at intervals, usually irregular, and functions normally at other times. It is caused by several contributing factors, some of which may be effectively random, which occur simultaneously.

(iv) Observability of the faulty model and their impact on diagnosis. The problem of identifying the correct set of measurements such that the system is diagnosable as well as observable, is quite an interesting research issue.

(v) Control Problem. In this thesis, we focused on the task of fault diagnosis. However, in many practical cases we are not interested in fault diagnosis by itself but rather as a part of control system. We want to take actions to fix them or at the very least minimize the damage that they cause, not just only identifying the true faults. Thus, there is an obvious need for a control component that would work hand in hand with our fault diagnosis tools.

### V.3 Conclusion

We have demonstrated in this thesis that dynamic Bayesian networks are a powerful tool for reasoning about complex and realistic domains. Combining the explicit representation of uncertainty that has proved useful for Bayesian networks with enough expressive power to model the continuous phenomena in hybrid domains. In this thesis, we apply the framework in mechanical domain and hydraulic domain, but it is not so hard to come up with other domains that could call for such diagnosis models, such as visual tracking, speech recognition, robotics and many others. In conclusion, the question of application of DBN model based diagnosis scheme is primarily a question of the quality of the available mathematical model of the system. In additional to this, the reachable quality of fault isolation and identification decisively depends on the available measurements.

## BIBLIOGRAPHY


Alonso-Gonzalez, C., Moya, N., and Biswas, G. (2010). Factoring dynamic bayesian networks using possible conflicts. *21st International Workshop on the Principles of Diagnosis*.

Arulampalam, S., Maskeel, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*.

Bice, D. (2001). Using stella models to explore the dynamics of earth systems: experimenting with earth's climate system using a simple computer model. *Journal of Geoscience Education*.

Biswas, G., Manders, E., Ramirez, J., Mahadevan, N., and Abdelwahed, S. (2004). Online model-based diagnosis to support autonomous operation of an advanced life support system. *Habitat.: Int. J. Human Support Res.*, pages 21–38.

Boyen, X. and Koller, D. (1998). Tractable inference for complex stochastic processes. *Proc. UAI*.

Broenink, J. (1999). Introduction to physical systems modeling with bond graphs. *SiE whitebook on Simulation Methodologies*.

Chen, J. (1994). A probabilistic theory of model-based diagnosis. *Int. J. Human-Computer Studies*.

Console, L. and Dressler, O. (1999). Model-based diagnosis in the real world: lessons learned and challenges remaining. *IJCAI'99 Proceedings of the 16th international joint conference on Artificial intelligence*.

Darwiche, A. (2001). Constant space reasoning in dynamic bayesian networks. *Intl. J. of Approximate Reasoning*.

de Kleer, J. and Kurien, J. (2003). Fundamentals of model-based diagnosis. *SafeProcess*.

Dearden, R. and Clancy, D. (2002). Particle filter for real time diagnosis of planetary rovers. *International workshop on Principle of Diagnosis 2002, DX02*.

Dijk, J. (1994). *On the role of bond graph causlity in modeling mechatronic systems*. PhD thesis, University of Twente.

Ding, S. (2008). *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer.

Dojer, N., Gambin, A., Mizera, A., Wilczyski, B., and Tiuryn, J. (2006). Applying dynamic bayesian networks to perturbed gene expression data. *BMC Bioinformatics*.

Duffield, B. and Hanford, A. (2002). Advanced life support requirement document. *Tech. Rep. JSC-38571 NASA Johnson Space Center, Houston*.

Forbes, J., Huang, T., Kanazawa, K., and Russell, S. (1995). The batmobile: Towards a bayesian automated taxi. *Proc. IJCAI*.

Francisco, J. and Marek, D. (2003). Reasoning under uncertainty. *Encyclopedia of Cognitive Science*.

Gertler, J. (1998). Fault detection and diagnosis in engineering systems. *Technology and Engineering, CRC Press*.

Gustavo, A., Enrique, L., and Villavicencio, A. (1996). Temporal reasoning under uncertainty for industrial diagnosis. *Proceedings ISAI/IFIS 1996. Mexico-USA Collaboration in Intelligent Systems Technologies*.

Jebara, T. (2005). Dynamic bayesian networks for multimodal interaction. *2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*.

Kabore, P., Staroswiecki, M., and Wang, H. (1999). On parity space and observer-based approaches for fdi in state affine systems. *38th IEEE Conference on Decision and Control*.

Kafai, M. and Bhanu, B. (2011). Dynamic bayesian networks for vehicle classication in video. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*.

Kawahara, Y., Yairi, T., and Machida, K. (2005). Diagnosis method for spacecraft using dynamic bayesian networks. *Proc. of The 8th International Symposium on Artifical Intelligence, Robotics and Automation in Space*.

Koller, D. and Lerner, U. (2000). Sampling in factored dynamic systems. *Sequential Monte Carlo Methods in Practice*.

Kortenkamp, D. and Bell, S. (2003). Biosim: An integrated simulation of an advanced life support system for intelligent control research. *Proceedings of the 7th Intl. Symp. on Artificial Intelligence, Robotics and Automation in Space*.

Lehner, P. and Sadigh, A. (1991). Reasoning under uncertainty: Some monte carlo results. *Proceedings of the Seventh Annual Conference on Uncertainty in Artificial Intelligence*.

Lerner, U., Parr, R., Koller, D., and Biswas, G. (2000). Bayesian fault detection and diagnosis in dynamic systems. *Proceedings of AAAI*.

Mihajlovic, V. and Petkovic, M. (2001). Dynamic bayesian networks: A state of the art. *CTIT technical reports series, Vol. TR-CTIT-34*.

Mosterman, P. and Biswas, G. (1999). Diagnosis of continuous valued systems in transient operating regions. *IEEE Transactions on Systems, Man, and Cybernetics*.

Murphy, K. (2002). *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley.

Narasimhan, S., Dearden, R., and Benazera, E. (2004). Combining particle filters and consistency-based approaches for monitoring and diagnosis of stochastic hybrid systems. *Proceedings of the Workshop on the Principles of Diagnosis 2004*.

Pickering, K., Wines, K., Pariani, G., Franks, L. A., Yeh, J., Finger, B. W., Campbell, M., Verostko, C., Carrier, C., Gandhi, J., and Vega, L. (2001). Early results of an integrated water recovery system test. *Proceedings of 29th Int Conf Environmental Sys*.

Pitt, M. and Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters. *J. Amer. Statist. Assoc*, 94(446):590–599.

Roychoudhury, I. (2009). *Distributed Diagnosis of Continuous Systems: Global diagnosis through local analysis*. PhD thesis, Vanderbilt University.

Roychoudhury, I., Biswas, G., and Koutsoukos, X. (2010). Distributed diagnosis of dynamic systems using dynamic bayesian networks. *18th Mediterranean Conference on Control and Automation*.

Russell, S. and Norvig, P. (2010). *Artificial Intelligence A Modern Approach - Third Edition*. Pearson.

Struss, P., Provan, G., de Kleer, J., and Biswas, G. (2010). Special issue on model-based diagnostics. *IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans*, 40(5).

Szarka, T. (2011). Structural, behavioral and functional modeling of cyber-physical systems. Master's thesis, Vanderbilt University.

Weng, J. and Biswas, G. (2012). A particle filter approach of tracking and estimating fault on nonlinear dynamic system. In *Proceedings of 23rd workshop on the principles of diagnosis*.

Zhang, Y. and Ji, Q. (2006). Active and dynamic information fusion for multisensor systems with dynamic bayesian networks. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICSPART B: CYBERNETICS*.

Zou, M. and Conzen, S. (2005). A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*.

Zweig, G. (1996). A forward-backward algorithm for inference in bayesian networks and an empirical comparison with hmms. Master's thesis, U.C. Berkeley.