Efficient Estimation for Diagnosis Using Factored Dynamic Bayesian Networks ⋆

Indranil Roychoudhury Gautam Biswas Xenofon Koutsoukos

Institute for Software Integrated Systems, Department of EECS, Vanderbilt University, Nashville, TN, USA 37235, {indranil.roychoudhury,gautam.biswas,xenofon.koutsoukos}@vanderbilt.edu

Abstract: Robust and efficient estimation of hidden state variables of a system in the presence of measurement noise and modeling errors is crucial for online model based fault diagnosis of continuous systems. Dynamic Bayesian Networks (DBNs) provide generalized and systematic methods for reasoning under uncertainty. This paper presents an approach to improve estimation efficiency by partitioning the DBN into smaller factors and invoking estimation algorithms on each factor independently. The factors are generated by replacing some state variables with algebraic functions of some measurement variables, thus reducing the across-time links between these state variables. Hence, given the measurements, these state variables become conditionally independent of the state variables in other factors, and the states of each factor can be estimated separately. This paper derives an algorithm for generating these factors and presents experimental results to demonstrate the effectiveness of our factoring approach for accurate estimation of system behavior.

Keywords: Statistical methods; Dynamic Bayesian Networks.

1. INTRODUCTION

Online detection and diagnosis of faults in engineering systems is essential for guaranteeing their safe, reliable, and efficient operation. Model-based diagnosis (MBD) schemes, such as [Mosterman and Biswas, 1999] and [Lerner et al., 2000], are preferred because they are more general than heuristic and data driven schemes. But, modeling inaccuracies and sensor noise must be accounted for before MBD schemes can be applied to real-world diagnosis problems. Probabilistic models, such as Dynamic Bayesian Networks (DBNs) [Murphy, 2002], exploit the conditional independence between system parameters and variables to provide a compact representation for reasoning about dynamic system behavior under uncertainty.

Diagnosis using DBNs involves solving the problem of estimating the hidden state variables in a DBN model at a time, t, using all measurement samples available up to t [Murphy, 2002]. [Lerner et al., 2000] presents an approach for the detection, isolation, and identification of abrupt and incipient faults by estimating nominal and faulty state variables. Exact state estimation is exponential in the number of state variables in a DBN. Moreover, for nonlinear systems and noise models with non-Gaussian distributions, analytic, closed form solutions for computation of the posterior probabilities may not exist [Murphy, 2002]. Approximate estimation algorithms, such as the Boyen-Koller (BK) algorithm [Boyen and Koller, 1998], and the particle filter (PF) [Koller and Lerner, 2001], mitigate this problem to some extent. In [Roychoudhury et al., 2008], the authors combined a qualitative fault isolation scheme (that employs symbolic analysis of measurement transients caused by faults) with a PF approach to reduce the complexity of the overall diagnosis task. PF approaches generalize traditional Kalman filter and extended Kalman filter approaches, since it can be applied to models with nonlinearities and arbitrary (non-Gaussian) probability distributions [Murphy, 2002].

This paper presents an approach for further increasing the estimation efficiency by partitioning the system DBN into multiple non-overlapping *DBN factors* (DBN-Fs) and invoking estimation algorithms on each DBN-F independently. The DBN-Fs are generated by replacing some of the state variables in the system DBN with algebraic functions of measurements (now considered as system inputs). Hence, the across-time links between these state variables can be removed. As a result, state variables in one DBN-F become conditionally independent of the state variables in all other DBN-Fs, given the measurements used to compute the values of the replaced state variables. This conditional independence between the state variables in each DBN-F serves as the basis for our truly distributed estimation approach, where estimation algorithms are invoked on each DBN-F independently, while only communicating some observed measurements between the factors. This distributed approach reduces a large exponential estimation problem to a set of smaller problems, and thus provides the framework for the gain in efficiency.

It is well-known that the state variables of a system can be estimated from the system measurements only if the system is *observable*. Traditional schemes of observability analysis apply only to linear systems. To extend the observability analysis to nonlinear systems, we adopt a methodology for *structural observability* analysis that is

⋆ This work was supported in part by the National Science Foundation under Grant CNS-0615214 and NASA NRA NNX07AD12A.

derived from a system's bond graph model [Dauphin-Tanguy et al., 1999]. Our partitioning scheme ensures that every DBN-F models a structurally observable subsystem. Hence, our distributed estimation scheme using DBN-Fs produces results equivalent to those obtained using the global DBN.

This paper presents an algorithm to systematically partition a system DBN into structurally observable DBN-Fs, and describes how estimation efficiency can be improved by applying a PF-based estimation scheme on each DBN-F independently. Experimental results demonstrate how the partitioning approach improves the efficiency of estimation without sacrificing accuracy.

2. RELATED WORK

Distributed decentralized extended Kalman filter (DDEKF) [Mutambara, 1998] is an exact estimation scheme which, like our distributed approach, subdivides the estimation problem into smaller problems. However, in DDEKFs, each local component requires measurements and estimates of state variables from other components to correctly estimate its states. As a result, the estimation algorithm must be invoked concurrently on all subsystems to ensure correct state estimates. On the other hand, by construction, the state variables in a DBN-F are conditionally independent of the state variables in all other factors, given some measurements. So, our estimation scheme is truly distributed since an estimation algorithm can be invoked on each DBN-F independently, since the estimation of state variables in one component does not depend on the state estimates at other components. Also, the failures in individual factors do not affect the estimates made at other factors as long as the required measurements are available.

The BK algorithm, presented in [Boyen and Koller, 1998], creates factors by eliminating causal links between weakly interacting subsystems, and represents the belief state as a product of these smaller factors. This removal of causal links imply that the factored belief state is necessarily an approximation of the complete belief state, and introduces error in state estimates. The authors prove this error to be bounded, but these bounds may not be sufficient to eliminate delayed or missed fault detection and isolation. In our factoring approach, we do not arbitrarily eliminate causal dependencies. We eliminate the across-time causal links to some state variables only if these state variables can be computed in terms of some measurements. Hence, our factoring scheme is not an approximation, and we preserve the dynamics of the unfactored DBNs.

[Frogner and Pfeffer, 2008] presents heuristic techniques for automatically decomposing a DBN into factors. This results in lower estimation errors, but the computed factored belief state is still an approximation. The Factored Particle Filtering (FPF) scheme [Ng and Peshkin, 2002] further reduces estimation errors by applying the PF scheme to the BK factored estimation approach. Our approach to factoring is based on analyzing the structural observability of the system's BG model. Because our factoring scheme preserves the overall system dynamics in the factored form, and every DBN-F is structurally observable by design, our PF-based estimation scheme produces accurate state estimates efficiently.

3. DYNAMIC BAYESIAN NETWORKS

A DBN is a compact representation of a Markov process and can be represented as $D = (\mathbf{X}, \mathbf{U}, \mathbf{Y})$, where $\mathbf{X}$, $\mathbf{U}$, and $\mathbf{Y}$ are sets of stochastic random variables that denote hidden (or state) variables, system input variables, and measured variables in the dynamic system, respectively [Murphy, 2002]. A DBN is a two-slice Bayesian network, representing a snapshot of system behavior in two consecutive time slices, t and $t+1$. State variables in the DBN model satisfy the first order Markov property, i.e., $P(\mathbf{X}_{t+1}|\mathbf{X}_t, \mathbf{U}_t)$, which is derived from the causal links $X_t \rightarrow X_{t+1}$, $X_t \rightarrow X'_{t+1}$, and $U_t \rightarrow X_{t+1}$, where $X', X \in \mathbf{X}$ and $U \in \mathbf{U}$, and subscript t represents a variable at time t. Similarly, the DBN observation model, $P(\mathbf{Y}_t|\mathbf{X}_t, \mathbf{U}_t)$, is derived from causal links, $X_t \rightarrow Y_t$ and $U_t \rightarrow Y_t$, where $Y \in \mathbf{Y}$. Therefore, the state estimation problem is defined as finding $P(\mathbf{X}_{t+1}|\mathbf{Y}_{0:t+1}) = \alpha P(\mathbf{Y}_{t+1}|\mathbf{X}_{t+1}, \mathbf{U}_t) \times \sum_{\mathbf{X}_t} P(\mathbf{X}_{t+1}|\mathbf{X}_t, \mathbf{U}_t) P(\mathbf{X}_t|\mathbf{Y}_{0:t})$, where $\mathbf{Y}_{0:t}$ denotes measurement readings from time 0 to t, and α is the normalizing factor [Murphy, 2002].

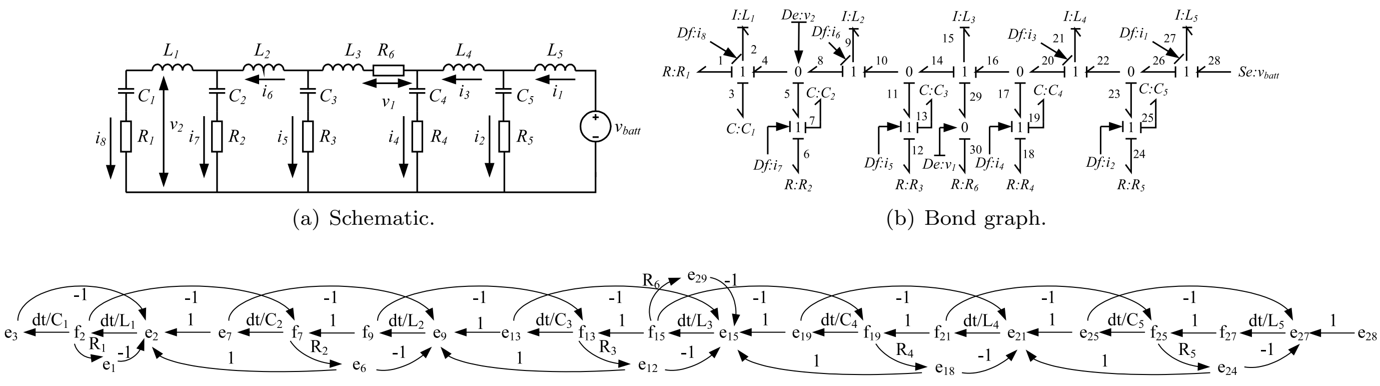
3.1 Deriving DBNs for Physical Systems

We have developed an approach for deriving DBNs from the temporal causal graph (TCG) models of physical systems [Lerner et al., 2000]. The TCG model is systematically derived from the bond graph model of the system [Mosterman and Biswas, 1999]

The bond graph (BG) modeling paradigm provides a framework for domain-independent, energy-based, topological modeling of physical processes [Karnopp et al., 2000]. The nodes of a BG include energy storage (capacitors, C, and inertias, I), dissipation (resistors, R), transformation (gyrators, GY, and transformers, TF), source (effort sources, Se, and flow sources, Sf), and detection (effort detectors, De, and flow detectors, Df) elements. Nonlinear systems are modeled using parameter values that are functions of other system variables. *Bonds*, drawn as half arrows, with associated effort, e, and flow, f, variables, represent the power interaction pathways between the bond graph elements, such that $e \times f$ defines the power transferred through the bond. 0- and 1-junctions represent idealized connections for lossless energy transfer between two or more BG elements.

Fig. 1(b) shows the BG model of an electrical circuit shown in Fig. 1(a). In the electrical domain, I elements are inductors, C elements are capacitors, R elements are resistors, flows represent current, e.g., f_2 denotes the current through L_1, and efforts represent voltage differences, e.g, e_3 denotes the voltage across capacitor C_1. The observed measurements in the electric circuit are the currents $i_1, i_2, \ldots, i_8$ and the voltages v_1 and v_2. The battery, v_{batt} drives this circuit. In this system, the BG parameters are assumed to be constant.

The *temporal causal graph* (TCG) of a system, systematically derived from its BG [Mosterman and Biswas, 1999], captures the causal and temporal relations between system variables through directed *edges* and their *labels. Causality* establishes the cause and effect relationships between the e and f variables of the bonds determined by constraints imposed by the incident BG elements. Of special interest



(a) Schematic. (b) Bond graph.

(c) Temporal causal graph.

Fig. 1. Different representations of an electrical circuit.

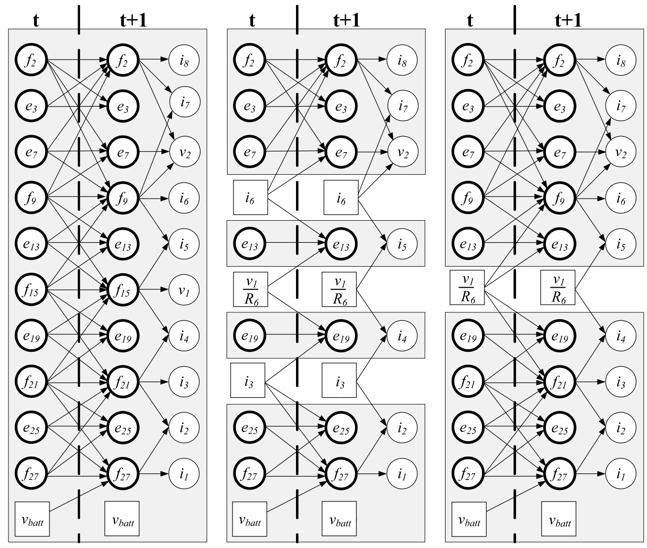
are the energy storage elements, which can either impose *integral* (preferred) or *derivative* causality. The sequential causal assignment procedure (SCAP) systematically assigns the causality in a BG [Karnopp et al., 2000]. The nodes in a TCG correspond to the power variables of the system BG model. Fig. 1(c) shows the TCG for the electrical circuit. The direction of a TCG edge and its label are based on causality. For example, for a C element in integral causality, $e = (1/C)\int f dt$, and hence the TCG edge directed from the flow to the effort has a label dt/C, with dt denoting a temporal relationship between f and e. For a C element in derivative causality, the TCG edge is directed from the effort to the flow, since $f = Cde/dt$, and has a label C/dt.

We construct the system DBN from its TCG in integral causality using the method outlined in [Lerner et al., 2000]. After we identify the TCG nodes, $\mathbf{N}$, which include all state variables, measured variables, and system inputs; for each $N \in \mathbf{N}$, we instantiate nodes N_t and N_{t+1} in the consecutive time slices of the DBN. Then, for every pair of variables, $N, N' \in \mathbf{N}$ that are algebraically related, causal links $N_t \to N'_t$ and $N_{t+1} \to N'_{t+1}$ are constructed in each DBN time slice. For every pair of variables, $N, N' \in \mathbf{N}$ having an integrating relation (i.e., a delay), the across-time $N_t \to N'_{t+1}$ link is added to the DBN. Fig. 2(a) shows the DBN for the electrical circuit, where thick-lined circles denote state variables, thin-lined circles denote observed variables, and squares denote input variables.

3.2 Observability

Correct estimation of dynamic system behavior is possible only if the system is *observable*, i.e., the values of all the state variables in the system can be correctly estimated from the knowledge of the system inputs and outputs [Samantaray and Bouamama, 2008]. Therefore, given a state-space representation of the system, with $\dot{\mathbf{X}} = f(\mathbf{X}, \mathbf{U})$, and $\mathbf{Y} = g(\mathbf{X}, \mathbf{U})$, a system is observable if there exists a well-behaved function, h, such that, $\mathbf{X} = h(\mathbf{Y}, \mathbf{U})$, i.e., there exists $|\mathbf{X}|$ independent equations which can be solved to correctly estimate the unknown $\mathbf{X}$. Since the variables in these equations are random variables, the system of equations still represent a stochastic process.

Existing numerical approaches for determining observability of a system are mostly applicable to linear systems.



(a) Full DBN. (b) 4-factored DBN. (c) 2-factored DBN.

Fig. 2. Factorings of the DBN of the electrical circuit.

The analysis of *structural observability* provides an elegant approach to determine system observability by analyzing the system bond graph model, rather than the system parameter values [Dauphin-Tanguy et al., 1999]. A BG is structurally observable if the following properties are satisfied:

(1) In the preferred integral causality mode, there exists at least one causal path from each I and C element in integral causality to a sensor element De or Df.
(2) Inverting the causality of *every* I and C element initially in integral (preferred) causality still produces a valid causal assignment for the entire BG [1].

In integral causality, the independent variables of C and I elements are the state variables of the system [Karnopp et al., 2000], and causal paths are directed from the state variables to the measured variables. Therefore, a system's failure to satisfy Property 1 implies that for a subset of the state variables, $\mathbf{X}_{ur} \subseteq \mathbf{X}$, which will not influence the state estimation process, since $P(\mathbf{Y}|\mathbf{X}) = P(\mathbf{Y}|\mathbf{X} - \mathbf{X}_{ur})$. In [Dauphin-Tanguy et al., 1999], the authors have proven

[1] In some situations, this may require changing a De or Df element into their dual form.

that the satisfaction of Property 2 establishes that the function h, such that $\mathbf{X} = h(\mathbf{Y}, \mathbf{U})$, exists.

Since DBNs can be systematically derived from TCGs generated from BGs, once a system is determined to be observable through structural analysis of a system BG, a DBN can be derived from this BG and a PF scheme can be invoked on this DBN to accurately estimate its state variables. We term a DBN to be observable if it represents an observable system.

4. FACTORING A DBN FOR EFFICIENT ESTIMATION

The basic idea of our factoring procedure involves identifying state variables in the complete DBN system model whose values are algebraic functions of at most r measurements, where r is a user-specified parameter. If a state variable can be algebraically computed from some measurements, we can replace that state variable with an algebraic function of these measurements, and remove the across-time links involving the replaced state variable. The removal of the across-time links enables partitioning of the system DBN into smaller DBN-Fs, such that the state variables in each DBN-F is *conditionally independent* of the state variables in all other DBN-Fs, given the measurements used to compute the value of some state variables. The objective of our factoring scheme is to apply this substitution repeatedly to remove as many across-time links as possible to generate DBN-Fs that allow accurate inferencing of system behavior when a estimation algorithm is applied to each DBN-F separately. Therefore, we need to ensure that every DBN-F in our factoring is observable. We generate the most number of observable DBN-Fs from a given system DBN through a two-step procedure: (i) generate maximal number of factors possible by replacing every state variable which can be determined as a algebraic function of at most r measurements, and (ii) *merge* unobservable DBN-Fs with other factors till all of the resultant factors are observable.

Consider the DBN shown in Fig. 2(b). If we assume $r = 1$, we can determine the value of state variable f_{15} as an algebraic function of voltage v_1, i.e., $f_{15} = h(v_1) = v_1/R_6$. Therefore, as shown in Fig. 2(b), we replace f_{15} with the input current v_1/R_6. Since, we no longer need the variables $f_9, e_{13}, f_{15}, e_{19}$, and f_{21} to compute f_{15}, the across-time links to f_{15} can be removed. Similarly, we can trivially determine the value of f_9 and f_{21} in terms of measurements i_6 and i_4, i.e., $f_9 = i_6$ and $f_{21} = i_3$, and hence, remove all across-time links to f_9 and f_{21}. Thus we generate four factors for the DBN of the electrical circuit. For this example, the two middle DBN-Fs are not observable, since the single state variable in either of the two DBN-Fs does not affect the observed variable, thereby violating Property 1. However the factoring generated by merging each unobservable DBN-F to its observable neighbor (see Fig. 2(c)) results in a factoring where all DBN-Fs are observable.

As outlined in Section 3.1, DBNs can be systematically derived from the BG system models. Also, observability of a system can be determined by structural analysis of its BG. Therefore, our algorithm for generating maximal number of observable DBN-Fs from a given DBN-F is as follows: (i) partition the system DBN into maximal possible DBN-Fs, (ii) map each generated DBN-F to a BG factor (BG-F) and analyze the structure of this BG-F to determine if the DBN-F is observable, and (iii) merge every unobservable DBN-F with other DBN-Fs till all DBN-Fs are observable. These steps are shown in Algorithm 1, and presented in detail below. We assume the unfactored system DBN to be observable, as otherwise, no factoring with only observable DBN-Fs exist. Also, we assume we have sufficient sensors to perform factoring.

Algorithm 1 Generating factors of a DBN.

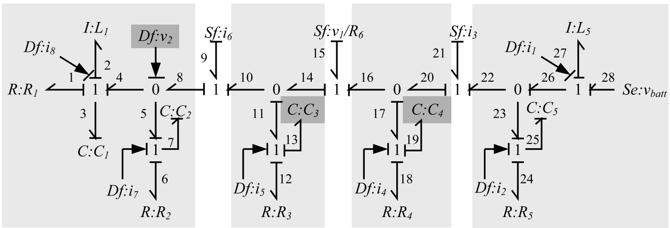
Input: System DBN, D
Generate maximal $Factoring_1 = \{D_1, D_2, \ldots, D_n\}$
$SetOfFactorings = \{Factoring_1\}$
while true do
  $SetOfObsF = \varnothing$; $SetOfUnobsF = \varnothing$;
  for each $Factoring_i \in SetOfFactorings$ **do**
    if every DBN-F in $Factoring_i$ is observable **then**
      $SetOfObsF = SetOfObsF \cup Factoring_i$
    else
      $SetOfUnobsF = SetOfObsF \cup Factoring_i$
  if $SetOfObsF \neq \varnothing$ **then**
    $BestFactoring = Factoring_j \in SetOfObsF$ having the most number of balanced DBN-Fs
    exit
  else
    $NextBestFactoring = Factoring_j \in SetOfUnobs$ having the most number of unobservable DBN-Fs
  $SetOfFactorings$ = all possible pairwise merging of the DBN-Fs of $NextBestFactoring$

4.1 Generating Maximal Factoring

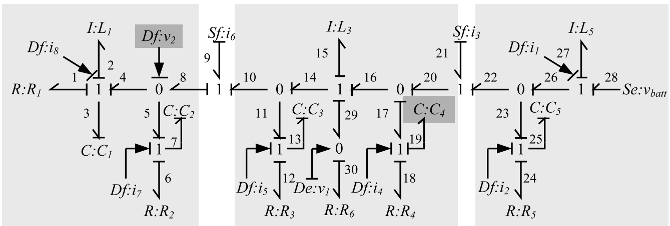
Given the parameter r, we analyze the system DBN to identify all state variables that can be computed from single measurements, then pairs, triples, and so on, up to r measurements. For example, if we consider $r = 1$, we can express $f_{15} = v_1/R_6$, $f_2 = i_8$, $f_9 = i_6$, and so on. However, we do not replace a state variable if the replacement does not generate any new factors. For example, in the maximally factored DBN shown in Fig. 2(b), replacing f_2 with i_8 does not generate any new factors, and hence, f_2 is not replaced.

4.2 Testing Observability of DBN-Fs

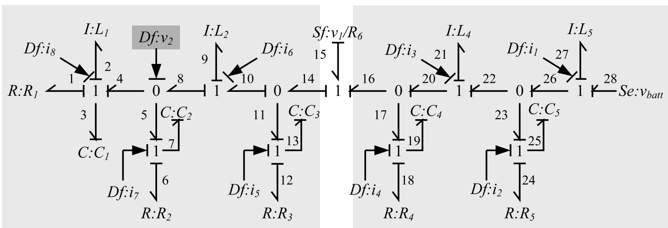
Once the maximal number of DBN-Fs are generated, for every DBN-F, D_i, we generate a corresponding BG-F, B_i, and consider D_i to be observable if B_i is analyzed to be structurally observable. Each B_i is constructed by replacing every I or C element that corresponds to a state variable that was replaced in the system DBN, with a Se or Sf element, each modulated by at most r measurements. Effectively, each replaced energy source creates multiple independent subsystems, thus factoring the BG into independent factors. Fig. 3(a) shows the BG-Fs corresponding to the 4-factored DBN shown in Fig. 2(b). The two outer BG-Fs are observable, since their energy storage elements can be assigned preferred derivative causality (albeit by dualizing an effort sensor into a flow sensor, indicated by the shaded background, in the first BG-F), and every state variable affects at least one sensor. The two BG-Fs in the middle, however, are not



(a) 4-Factored BG.



(b) 3-Factored BG.



(c) 2-Factored BG.

Fig. 3. Factorings of the BG of the electrical circuit.

observable, since, the state variable in neither of the two BG-Fs reach the flow sensor (whose value is determined by the two flow sources on the 0-junction). Since all BG-Fs are not observable in the maximal factoring, this factoring cannot be used for correct estimation of states.

4.3 Merging Unobservable DBN-Fs

An unobservable DBN-F can be *merged* with other DBN-Fs to generate an observable DBN-F. n DBN-Fs, $D_1, D_2, \ldots, D_n$, are merged by restoring the state variables that were replaced to generate $D_1, D_2, \ldots, D_n$, redrawing the across-time links causal links involving these restored state variables, and reintroducing the nodes corresponding to the measurements that were used to compute these state variables.

As shown in Algorithm 1, the merging procedure is invoked if every DBN-F in the maximally factored DBN is not observable. At every iteration step, we create new factorings through all possible pairwise merging of unobservable DBN-Fs into other DBN-Fs, with the goal of creating at least one new factoring with all its DBN-Fs observable. If multiple such factorings get created, we choose from amongst them that factoring which has the most number of *balanced* DBN-Fs, determined by comparing how close the number of state variables in each of its DBN-F is to the average number of state variables per DBN-F. If the merging step does not generate any factorings with all its DBN-Fs observable, we select the maximal factoring with the largest number of factors and highest number of unobservable DBN-Fs, and generate the next set of factorings by pairwise merging of unobservable DBN-Fs like before. This procedure is repeated till we obtain at least one factoring having only observable DBN-Fs.

No. of Factors →	1	2	4
Run 1	0.1591	0.1107	0.2034
Run 2	0.0681	0.1835	0.2389
Run 3	0.1215	0.0805	0.2495
Run 4	0.083	0.0952	0.1764
Run 5	0.0743	0.136	0.2121
Run 6	0.0788	0.2206	0.1518
Run 7	0.0974	0.1135	0.2206
Run 8	0.1581	0.1091	0.1671
Run 9	0.1556	0.2121	0.1659
Run 10	0.1467	0.1194	0.1824

Table 1. Average mean squared error over all state variables.

The unobservable DBN-Fs, shown in Fig. 2(b), can be merged in two possible ways, the corresponding BG-Fs of which are shown in Figs. 3(b) and 3(c). In Fig. 3(b), the BG-F at the center is generated by merging the two central BG-Fs in Fig. 3(a), and is not observable (since capacitor C_4 does not provide a consistent causal assignment when it is assigned derivative causality). However, the two factors in the factoring shown in Fig. 3(c) are observable, and hence, we select the DBN-Fs (shown in Fig. 2(c)) corresponding to these BG-Fs as our desired factoring.

5. ESTIMATION USING FACTORED DBNS

We choose PF as our estimation algorithm for estimating the values of state variables across time [Koller and Lerner, 2001]. A PF is a sequential Monte Carlo sampling method for Bayesian filtering that approximates the belief state of a system using a weighted set of samples, or particles [Arulampalam et al., 2002]. Each sample, or particle, consists of a value for each state variable, and describes a possible system state. As more observations are obtained, each particle is moved stochastically to a new state, and the weight of each particle is readjusted to reflect the likelihood of that observation given the particle's new state. The PF algorithm used in this work follows the one presented in [Koller and Lerner, 2001].

For estimating the states in n DBN-Fs $D_1, D_2, \ldots, D_n$, we distribute the estimation task amongst n PFs, each running as an independent process. Each of these PFs estimates $\mathbf{X}_i$ based on $\mathbf{Y}_i$ and $\mathbf{U}_i$. The PFs only communicate measurements $\bigcup_i \mathbf{U}_i$ between themselves. The PF for the DBN-F D_i uses $a\frac{|\mathbf{X_i}|}{|\mathbf{X}|}$ particles, where a is a user-specified parameter. Given n DBN-Fs, we know that $\sum_i |\mathbf{X_i}| < |\mathbf{X}|$, where $\mathbf{X}$ is the total number of states in the complete system DBN. Therefore, the complexity of estimating the states in each DBN-F is less that that of estimating the states in the global DBN. Also, since the estimation algorithms on the different factors are executed simultaneously, the complexity of estimation is determined by the complexity of estimating the states of the DBN-F having the largest number of state variables.

6. EXPERIMENTAL RESULTS

We present a set of experimental results to evaluate if estimation using factored DBNs can improve computational

No. of Factors →	1	2	4
Time (s)	137.03	37.74	18.97

Table 2. Time taken for PF to complete estimation.

efficiency without compromising accuracy. In this experiment, all prior and conditional probabilities are assumed to be Gaussian, and white Gaussian noise with zero mean and variance of 1 W was added to all the measurements.

For this experiment, we tracked the state variables of the three different DBN factorings shown in Fig. 2 for 10 runs. Given n DBN-Fs, $D_i = \{\mathbf{X}_i, \mathbf{U}_i, \mathbf{Y}_i\}$, $i = 1, 2, \ldots, n$, such that $\mathbf{X} = \mathbf{X}_1 \cup \mathbf{X}_2 \cup \ldots \mathbf{X}_n$, for each run we computed the estimation error: $E = \frac{1}{|\mathbf{X}|} \sum_{X \in \mathbf{X}} \left(\frac{1}{T} \sum_{t=0}^{T} (X_t - X_t^{model})^2 \right)$, where T is the total simulation time, X_t denotes the estimated value of state X at time t, and X_t^{model} denotes the actual value of state X at time t obtained from the simulation model. Table 1 reports the errors obtained from each factoring for all runs.

Our primary goal for this experiment was to demonstrate that the factoring scheme preserves the system dynamics. Therefore, we hypothesized that the difference in errors for the 2-factor and unfactored DBN would not be statistically significant, and the error for the 4-factor DBN would be significantly larger than the unfactored DBN. Further the difference in error for the 2-factor and 4-factor DBNs would also be statistically significant. We ran t-tests to establish significance of the differences. The tests for significance indicated that the errors obtained using the 2-factor DBN did not significantly differ from that obtained using the unfactored DBN ($p < 0.05$), while those obtained using the 4-factor DBN was significantly greater ($p < 0.05$). The test of significance between the 2- and 4-factor DBN showed that the error in the 4-factor DBN was significantly larger ($p < 0.05$). Therefore, we concluded that the 2-factor DBN preserves dynamics of the unfactored DBN, whereas the 4-factor DBN, which has unobservable factors, does not.

Table 2 shows the average time taken by the slowest PF for each factoring to track system behavior for 1500 time steps. The time taken by a PF depends on the number of particles it uses. For our experiments, the number of particles the PF for a factor used was proportional to the number of states in that factor. Hence, PF for unfactored DBN (with 1000 particles) took the most time, followed by the PF on the larger DBN-F of the 2-factor DBN (with 500 particles). The least amount of time was taken by the PFs applied to the 4-factor DBN, since its largest DBN-F has 3 state variables, and hence, its PF used 300 particles.

7. DISCUSSION AND CONCLUSIONS

In this paper, we presented an approach to increase the efficiency of estimation using DBNs by factoring the DBNs into DBN-Fs, such that the state variables in every DBN-F is conditionally independent from those in other DBN-Fs, given the measurements communicated between these factors, thus preserving the dynamics of the global system behavior. Experimental results showed that factoring maintains estimation accuracy in DBNs while improving the efficiency of DBN estimation in the presence of sensor noise. However, we need to evaluate the effect of sensor faults in our estimation, especially if the faults are in sensors that decouple two or more factors. Our intuition is that the presence of a fault in such a sensor will affect the estimation accuracy of the state variables in only those factors which uses this sensor as an input. The estimation accuracy of state variables in other factors will remain unaffected by this sensor fault, because of their conditionally independence from other state variables, given the measurements used as system inputs. Future work will therefore focus on applying this factoring scheme to develop distributed diagnosis approaches for complex, real-world systems, and evaluating estimation accuracy in the presence of sensor faults.

REFERENCES

M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. on Signal Processing*, 50(2):174–188, 2002. ISSN 1053-587X.

X. Boyen and D. Koller. Tractable inference for complex stochastic processes. In *Proc. of the* 14^{th} *Annual Conference on Uncertainty in Artificial Intelligence*, pages 33–42, 1998.

G. Dauphin-Tanguy, A. Rahmani, and C. Seur. Bond graph aided design of controlled systems. *Simulation Practice and Theory*, 7:493–513, 1999.

C. Frogner and A. Pfeffer. Discovering weakly-interacting factors in a complex stochastic process. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 481–488. MIT Press, Cambridge, MA, 2008.

D. C. Karnopp, D. L. Margolis, and R. C. Rosenberg. *Systems Dynamics: Modeling and Simulation of Mechatronic Systems*. John Wile & Sons, Inc., New York, NY, USA, 3^{rd} edition, 2000.

D. Koller and U. Lerner. Sampling in factored dynamic systems. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

U. Lerner, R. Parr, D. Koller, and G. Biswas. Bayesian fault detection and diagnosis in dynamic systems. In *Proc. of Seventeenth National Conference on Artificial Intelligence*, pages 531–537, 2000.

P. J. Mosterman and G. Biswas. Diagnosis of continuous valued systems in transient operating regions. *IEEE-SMCA*, 29(6):554–565, 1999.

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

A. Mutambara. *Decentralized Estimation and Control of Multisensor Systems*. CRC Press, 1998.

B. Ng and L. Peshkin. Factored particles for scalable monitoring. In *In Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 370–377. Morgan Kaufmann, 2002.

I. Roychoudhury, G. Biswas, and X. Koutsoukos. Comprehensive diagnosis of continuous systems using dynamic bayes nets. In *Proc. of the* 19^{th} *International Workshop on Principles of Diagnosis*, pages 151–158, 2008.

A. K. Samantaray and B. O. Bouamama. *Model-based Process Supervision: A Bond Graph Approach*. Springer, 2008.