Qualitative Bayesian Failure Diagnosis for Robot Systems

Dominik Kirchner[1] and Kurt Geihs[1]

Abstract— **Reliability is a key challenge for intelligent robot systems. In order to address this challenge, runtime failure detection and diagnosis (FDD) is an essential task to maintain autonomous operation. The complexity of fully fledged robot systems and the included noise in system observations complicate this task. In this paper, we present our Qualitative Bayesian Failure Diagnosis (QBFD) for precise and robust failure estimation. Our approach uses a Dynamic Bayesian Network to model uncertainties of the measurements while considering temporal relations. Instead of detailed a priori knowledge of system dynamics, our approach models cause-effect relations. These relations are, in practice, more intuitive to specify. As a consequence, we reduce the level of needed system knowledge and therefore increase the practical applicability. We evaluate the quality in respect to two reference approaches in extensive simulations. Due to our results, we are confident that our proposed approach provides comparable, if not superior, estimation quality, while simultaneously reducing the level of needed model details. Furthermore, we provide evidence that, given a proper system decomposition, high quality estimates are possible using general observations, like the resource usage.**

I. INTRODUCTION

Recent advances in the field of Artificial Intelligent (AI) yield to constantly improved and increasingly intelligent autonomous robot systems. As a consequence of the ongoing progress in the fields of robotics and AI, complexity of such robot designs grows fast. In fact, most of these systems have reached a level of complexity where manual administrative efforts of maintaining a reliable system are getting demanding and error-prone. Considering the main design objective, namely reliable autonomous operation in unstructured every day tasks, manual administrative efforts needs to be minimized. Hence, stable autonomous operation and reliability have to be considered as one of the major current challenges on the way to real world applications.

In order to contribute to this challenge, manual efforts of human operators should be reduced by combining reliability engineering with AI techniques to create an autonomous control mechanisms for detecting, diagnosing, and recovering system failures [1]. In such a design, high quality of failure detection and diagnosis (FDD) is essential. Indeed, correctness and robustness are main quality properties of FDD [2].

However, the combination of these techniques in a FDD approach faces multiple problems in the robot domain. Continuous system monitoring is an essential requirement for FDD. However, system monitoring is rarely initially included in the system architecture [3]. Moreover, a late integration often causes high integration efforts [4]. As a consequence, monitoring support is often omitted, where the integration efforts seems too high. In practice, the coverage of the monitored system state is limited. Therefore, late integration of monitoring tends to provide only a partial overview of the system state, often based on easily accessible general system information. Hence, the limited amount of available state information further complicates reliable failure estimation.

Furthermore, FDD often incorporate robot-specific knowledge of the system dynamics [5]. The system dynamics, usually represented in a model, describe the expected behavior of the system in a given situation. However, the complexity and uncertainty of physical systems, like robots, make a precise dynamic model of a complete system a difficult and time consuming task. In practice, comprehensive and precise knowledge representation in models are rarely available [6].

Additionally, operation of autonomous robots are computational expensive. However, FDD have to work in soft real-time to enable a timely recovery as well. Efficiency is, therefore, an essential requirement in FDD design.

In summary, FDD for robot systems have to deal with limited state information, imprecise knowledge of the system dynamics, and the requirement of minimal resource usage. In this paper, we present the Qualitative Bayesian Failure Diagnosis (QBFD) that is designed to cope with uncertainty, ignorance of the dynamics, and efficiency. Our approach is based on a Dynamic Bayesian Network (DBN) that is inherently able to deal with time-related and uncertain observations. The approach relies on restricted knowledge of cause-effect relations of single components instead of a dynamic model of the complete system. Therefore, the FDD problem is segmented to reflect the distinct components of the system, like object detection, decision making, localization, and so on. Each segment is dedicated to analyze one specific component. The resulting component-based scope of the FDD enables an intuitive way to specify component specific failures (causes) and consequent results (effects).

The rest of the paper is organized as follows: the next section describes related work. Section III describes the proposed approach, while Section IV presents experimental results. Simulation experiments have been performed in which related diagnostic approaches, discussed in Section II, are compared to the results of the presented approach. The paper concludes with a summary.

II. RELATED WORK

In general, diagnostic methods comprise of two basic components: a priori domain knowledge and search strategy [7]. In this section, we discuss FDD paradigms [2], [7], [8] based on their used a priori knowledge and place our

[1]Distributed Systems Group, Electrical Engineering and Computer Science, University of Kassel, Wilhelmshher Allee 73, 34119 Kassel, Germany {kirchner,geihs}@vs.uni-kassel.de

contribution in this context. The basic a priori knowledge is a set of defined failures and the relationship between the observations and these failures. A diagnostic system may have them explicitly, e.g. in a table, or it may be inferred from some source of domain knowledge. We distinguish this in explicit and model-based a priori knowledge.

Explicit knowledge is often derived from empirical knowledge or may be compiled from past experience with the system. Approaches based on past system data are referred to as process history-based knowledge [8] or data-driven [6], while methods that uses a predefined set of symptoms are called fault-signature approaches (FS) [9]. Here, a fault signature, or pattern, is a vector of symptoms for each defined failure. The goal is to find the best match of the current observations to a set of known symptom patterns for each possible failure. A search strategy formulated as a logical diagnosis rule is often used to realize the matching, as demonstrated in RoboCup by [10]. This common implementation of a FS approach has the advantage of an empirical configuration of the needed a priori knowledge and provides reliable diagnostic results. However, the quality of the results is highly application specific and depends on manual, subjective specification. As a consequence, observation variations and specification errors often impair robustness and correctness of the results.

Beside using explicit knowledge, a priori domain knowledge may be developed from a fundamental understanding of the system using first-principles knowledge. Such knowledge is referred to as model-based knowledge [11]. The model-based a priori knowledge can be further classified as qualitative or quantitative.

In quantitative models, this understanding is expressed in terms of mathematical functional relationships between the inputs and outputs of the system based on some fundamental insight of the dynamics of the system. This relation can be expressed in different mathematical models, as presented in [11], [7]. In the Multi Model Adaptive Estimation (MMAE) [5], Kalman filters are used to formulate this model. Here, each system failure is modeled as a separate Kalman filter. The residual between the estimates of a model and the real process behavior is used as an indicator for failure identification. Typically, the model with the minimal residual is chosen for the diagnostic estimate. Due to the properties of the Kalman filtering, the estimates are efficient in terms of resource usage and robust in respect to disturbances of the observations. However, the specification of the underlying process model is hard, requires expert knowledge of the system dynamics, and significant experience in filter design. Therefore, the performance of this approach is good, but the integration effort is considered high.

In qualitative models these relationships are expressed in terms of qualitative functions centered around distinct units in a system, like failure symptoms. The qualitative models can be developed either as qualitative causal models or abstraction hierarchies [7]. Different approaches exist to model the cause-effect relationship for these approaches, e.g. Fault Trees [12], signed digraphs [13], or quantitative physics. Though, qualitative models have a number of advantages, the major disadvantage is the generation of spurious solutions.

Our approach, as described in detail in Section III, follows the qualitative representation of the a priori knowledge. Instead of using dynamic knowledge of the system, we model cause-effect relations. These relationships are considered to be more intuitive to specify and, therefore, this should facilitate the development of the model. Uncertainties are inherently addressed in a DBN to counter the robustness problems of the FS. Furthermore, time relations, like in the process history-based methods are expressed in temporal relations in the DBN. Furthermore, we decompose the system in components to reduce the problem complexity.

III. FAULT DETECTION AND DIAGNOSIS

Our work focuses on a FDD design that incorporates the special requirements of the robot domain. This includes the capability to deal with uncertain noisy measurements and ignorance to a comprehensive and precise system model.

Many of the modern robot systems are modular and consist of multiple components with clearly defined and limited interfaces. Therefore, the dependencies between components are limited. To reduce the overall complexity of the approach, we subdivide the FDD problem in multiple separate tasks, one for each component.

While it is difficult to monitor intrinsic details of a component, it is much easier to observer multiple general characteristics. The operating system offers a wealth of general, but easily accessible, characteristics for software components, like CPU usage, memory usage, number of threads, and so on. Therefore, our approach combines multiple general component characteristics to determine reliable state estimates.

Instead of using dynamic knowledge, we propose a qualitative approach based on cause-effect relations. Therein, we only consider direct causal relations of assumed failures (causes) to the consequent effects. No comprehensive dynamic model needs to be specified. This knowledge is considered more intuitive and, hence, the modeling is expected to be easier.

The task of the FDD is to estimate the state of each system component separately. In order to conclude a global system state, the atomic estimates can be recombined in accordance to the system structure as found in [14]. A high quality estimate of each component is therefore essential.

In the following, we will focus on FDD for software components of a robot system. To develop such a component related FDD approach, we start to specify the problem description in a formal way to clarify the design goal.

A. Formal Problem Description

We assume that the robot system provides monitoring capabilities to observe at least a subset of the system's state information. Furthermore, we suppose that the monitoring system provides a finite set $\mathcal{O}$ of time-variant system observation $o_n(t_n)$, where $n \in \{1, ..., N\}$. A subset of these observations $\mathcal{O}_k$ is related to a specific component k,

$$\mathcal{O}_k := \{o_1(t_1), o_2(t_2), ..., o_N(t_N)\}. \quad (1)$$

Each of these observations have to be considered as a noisy measurement. These observations represent the current update information for the component k. The domain of the observations are expected to be real-valued $o_n \in \mathbb{R}$ and the number of elements in the set are assumed to be finite. The first task of the FDD is to detect a failure. Therefore, the FDD needs to estimate the current general component state $\mathcal{S}_k$.

$$\mathcal{S}_k := \{s_{ok}, s_{failure}\} \tag{2}$$

This set contains two states: the failure free state s_{ok} and the failure state $s_{failure}$. The second task of a FDD is the failure diagnosis. Failure diagnosis identifies root causes f_m of a failure. We assume that there exist only a finite set $m \in \{1, ..., M\}$ of possible root causes for a component k.

$$\mathcal{F}_k := \{f_1, f_2, ..., f_M\} \tag{3}$$

The challenge, which we have to address, is to develop a function fdd that maps observations $\boldsymbol{\mathcal{O}_k}$ to estimates of a state (detection) $\mathcal{S}_k$ and a set of root causes (diagnosis) $\boldsymbol{\mathcal{F}_k}$.

$$\text{fdd} : \boldsymbol{\mathcal{O}_k} \rightarrow \mathcal{S}_k \times \boldsymbol{\mathcal{F}_k} \tag{4}$$

B. Bayesian Failure Detection and Diagnosis

We develop the given problem in the framework of probability theory. In equation 4, our problem is defined by the estimates $s \in \mathcal{S}_k$ and $\boldsymbol{f} \in \boldsymbol{\mathcal{F}_k}$ based on a set of current observations $\boldsymbol{o} \in \boldsymbol{\mathcal{O}_k}$. We interpret these elements as random variables $(S, \boldsymbol{F}, \boldsymbol{O})$ of an atomic event in the joint probability distribution $P(S, \boldsymbol{F}, \boldsymbol{O})$. The estimation problem is now related to the determination of the atomic event's probabilities.

The monitoring system provides continuous updates of the system state. This is represented as evidence of the observations $\boldsymbol{O}$. Note that we do not require evidence for every observation, as incomplete evidence is sufficient. Based on the given evidence, the detection and diagnosis task can be reformulated as conditional probabilities:

$$P(S = s_n \,|\, \boldsymbol{O}) \qquad P(F = f_m \,|\, \boldsymbol{O}) \tag{5}$$

However, the inference of these atomic events can become intractably complex for a growing number of variables and is rather unnatural to specify [15]. An efficient way to describe a joint probability distribution is a graphical model, e.g. a Bayesian Network (BN). More specifically, we use a Dynamic Bayesian Network (DBN) to address the time of the observations $o_n(t_n)$. A DBN is a directed acyclic graph $G = (V, E)$, whose nodes $V := \{X_1, X_2, ...X_P\}$ are random variables X. The edges $E := \{e_1, e_2, ..., e_Q\}$ in the graph represent direct dependencies between the nodes $P(X_i|X_j)$. As before, $i \neq j$ are indices of the corresponding set. To include a priori domain knowledge, we exclude edges in the graph [15].

$$e_q = P(X_i|X_j) = P(X_i) \qquad e_q \notin E \tag{6}$$

In order to derive the structure of the graphical model, we discuss some assumption of the domain:

- **Single-Fault:** We assume that a component can only suffer one failure at a time. Therefore, we summarize all possible causes for failures (root causes) as states in one random variable FT. To complement the state probabilities to one, we add the state named f_{ok}.

$$P(F_1, ..., F_m) \rightarrow P(FT) =< f_1, ..., f_M, f_{ok} > \tag{7}$$

- **Observer-Independence:** Observations are modeled as separate nodes and assumed to be independent from each other. Edges between observations are excluded from the graph.

$$P(O_i \,|\, O_j) \rightarrow P(O_i) \tag{8}$$

- **Causality:** The observations O_i are modeled as a direct causal result from the root causes FT.

$$P(O_i \,|\, FT) \tag{9}$$

- **Direct-Diagnosis:** The states of the detection are modeled as direct consequences of the root causes. The sum of all root cause probabilities, excluding the normal state, determines the failure detection $s_{failure}$ probability.

$$P(S \,|\, FT) \tag{10}$$

As a result, we get a more compact structure with a reduced number of edges.

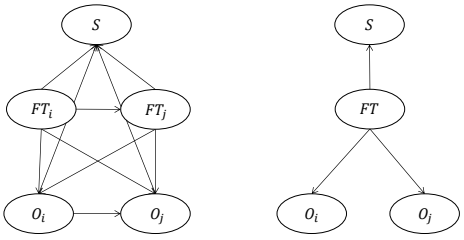

Fig. 1. left: fully meshed BN; right: BN with included domain knowledge

In order to model the dynamic progression of the observations, we define temporal transitions. Therefore, this compact graphical model needs to encompass both: the observation model (see Figure 1) and the transition model. The temporal behavior of a DBN is modeled in a finite set of discrete equidistant time steps $0 : t$. Thus, each random variable X has an instantiation for each time slice X_t. We continue to include domain knowledge in order to further reduce the complexity of the model:

- **Markov-Assumption:** In a Markov-Assumption random variables X_t only depend on random variables one time step earlier X_{t-1}. As specified, the state node S only depends on the root causes FT. As a consequence, the state node is independent from the last state estimate.

$$P(FT_t \,|\, FT_{0:t}) = P(FT_t \,|\, FT_{t-1}) \tag{11}$$

- **Markov-Sensor-Assumption:** In addition to the Markov-Assumption, we assume that the observation nodes $\boldsymbol{O_t}$ are independent in respect to time.

$$P(\boldsymbol{O_t} \,|\, FT_{0:t}, \boldsymbol{O_{0:t}}) = P(\boldsymbol{O_t} \,|\, FT_t) \tag{12}$$

These assumptions reduce the temporal model of the proposed DBN to one temporal transition.

$$P(FT_t \mid FT_{0:t}, \boldsymbol{O}_{0:t}) = P(FT_t \mid FT_{t-1}, \boldsymbol{O}_t) \tag{13}$$

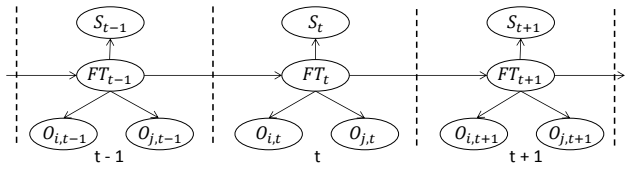

Fig. 2. Dynamic Bayesian Network for Fault Detection and Diagnosis

In order to complete the proposed DBN, we discuss the structure of the temporal transition $P(FT_t \mid FT_{t-1})$. In accordance with the Single-Failure-Assumption, a failure has to be resolved before a new failure can occur. Therefore, we prohibit direct transitions from one failure $FT_{t-1} = f_i$ to the next $FT_t = f_j$. This is represented as zeros in the transitional matrix.

$$P(FT_t \mid FT_{t-1}) = \begin{pmatrix} \theta_1 & 0 & \cdots & e_1 \\ 0 & \theta_2 & \cdots & e_2 \\ \vdots & \vdots & \ddots & \vdots \\ r_1 & r_2 & \cdots & \theta_{M+1} \end{pmatrix} \tag{14}$$

Here, the probability to stay in a given failure state $P(FT_t = f_m \mid FT_{t-1} = f_m)$ is denoted by θ_m, while the probability to recover from a failure $P(FT_t = f_{ok} \mid FT_{t-1} = f_m)$ is denoted by r_m. The failure occurrence probability $P(FT_t = f_m \mid FT_{t-1} = f_{ok})$ is given by e_m.

The probability of a successful recovery r and an enduring failure θ depend on the quality of diagnosis and recovery. There exist various failure models to estimate the failure occurrence probability [16]. These models are often based on the development process, like the testing time. Hence, values of these variables are domain specific and can not be specified in general.

Using Bayesian inference methods [15], we can infer the probability of all the states of S and FT. However, the FDD method needs to provide one distinct estimate for detection and diagnosis. In order to make the decision of the estimates, we apply a maximum a posteriori estimation (MAP) (shown for the detection task).

$$\hat{s} = argmax_S P(S|\boldsymbol{O}) = argmax_S \frac{P(\boldsymbol{O}|S)P(S)}{\int_S P(\boldsymbol{O}|S)P(x)\mathrm{d}x}$$

IV. EVALUATION

To demonstrate the efficiency of QBFD, we conducted simulations to evaluate the quality in terms of the properties: correctness and robustness. The correctness property is defined as the ratio of the correct estimates to all estimates. The robustness property, in addition, captures the behavior of the approach in respect to disturbances. In this context, disturbances are understood as uncertainties represented as noisy measurements or modeling errors due to wrongly specified model parameters.

In order to ground this evaluation, we compare our results with two reference approaches. One reference is a FS approach. The diagnostic rule is formulated in propositional logic and is designed to maximize the reliability of the estimates. As the second reference, we choose a model-based approach. Therefore, we implemented the MMAE approach [5], described in Section II.

The proposed QBFD builds on cause-effect relations modeled in a DBN and continuous state updates. While our evaluation focuses on the software layer of a robot system, we expect QBFD to be directly usable to robots hardware layer as well. Therefore, external sensors to observe the physical component state need to be integrated, like temperature, voltage, current, sound, and so on.

A. Simulation Setup

The setup of the experiment consists of a simulated robot system, a monitoring component, an implementation of the FDDs under evaluation, and a component to trigger predetermined failures. Beside the failure simulator, all components are basic parts of the RoSHA architecture [1]. RoSHA is designed to integrate failure recovery features for already existing robot systems.The simulation system consists of multiple components to reflect modular data processing and inter-component communication of most modern robots.

In order to observe current state information, we use the Capability and Reliability Manager (CARE) [17] for system monitoring. CARE provides adaptive and efficient monitoring services to limit monitoring resource overhead [3]. In this setup, the adaptation controls the observation frequency. We use no support of the components that are part of the simulated robot system. Therefore, the results, in this experiment setup, are purely based on general, component-independent monitoring information. Most of this information, like CPU are directly accessible from the operating system. Additionally, CARE monitors the inter-component communication to provide information on general communication characteristics, like message frequency or message content. The observed characteristics in this simulation are the CPU usage, memory usage, number of threads, message frequency, and console output frequency.

In the experiment, we focus on the analysis of one specific component of the system. In accordance to the RoSHA architecture, QBFD and the reference FDD approaches are implemented as a plug-in for a system-wide diagnostic system. Due to the decomposition of the system, each plug-in analyses one component. The implementation of QBFD is based on the SMILE library [18].

Finally, we use an additional component, the failure simulator, to trigger a set of predefined failures. In our experiment, we support failure types that are related to general processing faults, like blocking of the processing, uncontrolled processing, and component crashes. More specifically, our reference failures for these general fault types are deadlocks (DL), endless loops (EL), and crashes (NP). Faults in the algorithm of a component are case-specific and therefore not addressed in this simulation.

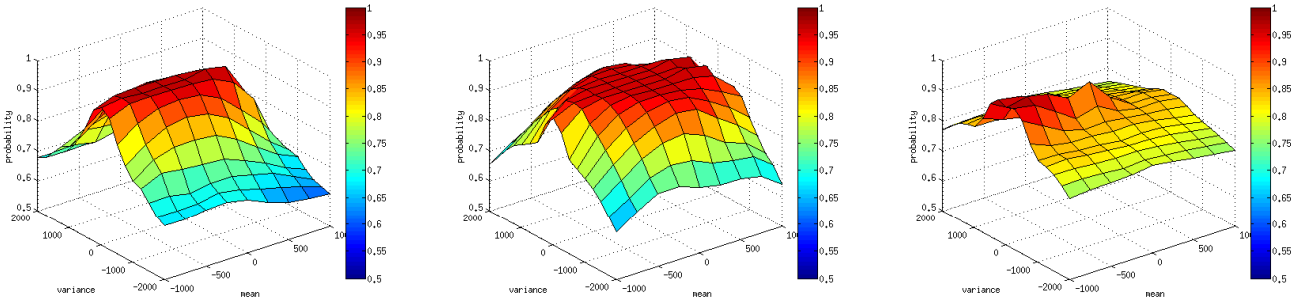

Fig. 3. Diagnosis results in respect to disturbance. left: FS approach; middle: QBFD; right: MMAE approach

B. Experiment Procedure

We design a run of an experiment to trigger alternately failure states and operational states. A run continues until all specified failure types are triggered, see Figure 4.

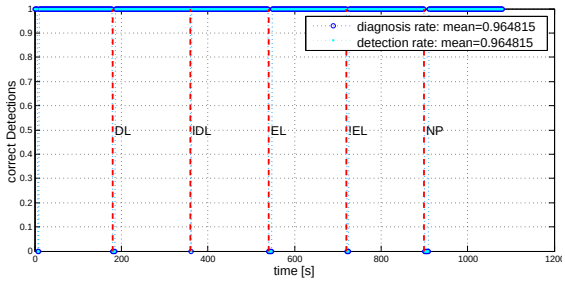

Fig. 4. QBFD results for run 1 with mean: 0 and variance: 0

Due to misleading former observations, the occurrence of wrong FDD results is concentrated directly after state transitions. Thus, we kept the time intervals between occurring failures (MTBF) short (3 minutes) to model a demanding evaluation scenario. More realistic practical MTBF values are around 12 hours [19], [20].

In order to determine the robustness property, we iterate over discrete values of the disturbed situation. For the FS reference approach, we need to discretize the continuous values to ground the logical symbols of the diagnostic rule. In order to ensure unified disturbance parameters, independent of concrete value ranges, we specify the disturbance model as a Gaussian distribution, where the mean and variance settings are modeled as factors in respect to the center and the width of the discretization interval. In our experiment, we vary the mean setting in the interval $[-1, -0.875, ..., 1]$ and the variance in the range $[0, 0.125, ..., 2]$. Furthermore, we conduct five runs to get a good average for each estimate.

C. Discussion of the Results

We represent the correctness and robustness properties for each approach in a separate plot, see Figure 3. High values of the correctness property present highly reliable estimates. A next to constant function curve implies a high degree of robustness for a given correctness level. Additionally, we present the combined results in a contour plot to provide a direct comparison. The border lines represent the values where the approaches drop below a certain level of quality. Figure 5 illustrates this for a 90% correctness property.

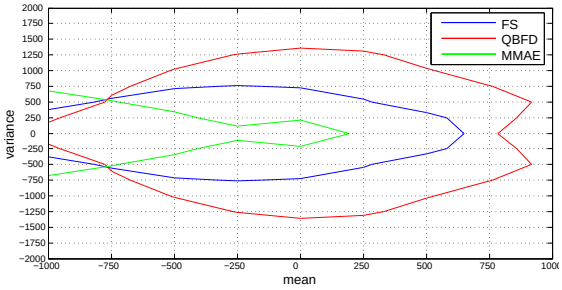

Fig. 5. Border lines for a 90% correctness quality requirement

All considered approaches provide high correctness values for the diagnostic task in a low disturbance configuration (96.05% (QBFD), 95.85% (MMAE), 96.39% (FS) for disturbance-free configuration). Results of the detection task in a disturbance-free configuration are nearly identical (96.04% (QBFD), 97.05% (MMAE), 96.39% (FS)). While in the FS and the QBFD the detection is a direct consequence of the diagnosis (see equation 10), and therefore provides, beside numerical deviations, similar results, the MMAE approach significantly improves its correctness property. As depicted in Figure 3, the FS approach seems robust against additional introduction of an offset (mean), while it is sensitive to the variance of the disturbance. The MMAE reacts more robust to high level disturbance. The underlying Kalman filters are designed to process noisy data and, therefore, are assumed to be the cause for that. However, the diagnostic correctness in a low disturbance setting is significant less (see Figure 3, right). This is a consequence of the adaptive monitoring. The adapted observation intervals may produce incomplete observation vectors, what impairs the MMAE performance. Simulations with unified observation intervals show higher correctness results.

QBFD is inherently more robust against incomplete observation evidences. Therefore, we experience high robustness (until a middle level of disturbance), while simultaneously providing high correctness values. The drop in highly disturbed areas is very likely to be caused by discretization effects in the grounding process of the discrete random variables. While the FS approach shares the same implemen-

tation of the discretization process, the temporal Recursive Bayesian Filtering seems to stabilize the estimates.

We evaluate our approach in an exaggerated setting with short state durations of 3 minutes, instead of several hours in a real-world setup. Extending the state duration, the results converge asymptotically to 1. see Figure 6. In experiments with a MTBF of 15 minutes, the correctness values increase to over 99%. Further increasing the MTBF to 60 minutes, we even exceed correctness values of 99.8%. The drop of the FS results is due to unexpected low memory usage observations. In contrary, MMAE and QBFD are robust to these disturbances.

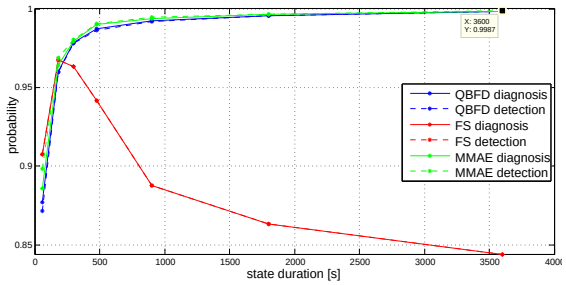

Fig. 6. FDD quality in respect to increasing state duration time

As a consequence, the presented results are two folded. While the MMAE approach provides the best properties for the detection task, the QBFD, due to its superior correctness, seems to be a good choice for the diagnostic task. Note, that the presented results are based on an adaptive monitoring [3]. Given a complete observation set, additional experiments have shown that the MMAE approach provides high quality for detection and diagnosis.

However, the MMAE relies on a detailed knowledge of the component dynamics. This specification is not trivial, nor intuitive. As a consequence, the implementation of such an approach is demanding, especially for complex and dynamic systems, like robots. In contrary, the QBFD uses cause-effect relations as a priori knowledge that are considered to be more intuitively to specify. Therefore, we argue that the integration process is facilitated, while ensuring adequate FDD results. In summary, the QBFD results are a good compromise between integration effort and estimation quality.

V. CONCLUSIONS

The task of failure detection and diagnosis faces special challenges in the robot domain, namely noisy observations and imprecise knowledge of the systems dynamics. In this paper, we present a FDD approach that addresses these challenges. We propose a FDD based on a Dynamic Bayesian Network to model both the inherent uncertainty of the domain and the temporal relation of the estimates. Furthermore, we decompose the robot system in atomic components and analyze each separately. As a consequence, the responsibilities of the FDD tasks are limited to one specific component and hence the complexity is reduced. Furthermore, our approach relies on cause-effect relations instead of dynamic a priori knowledge. These relations are claimed to be more intuitive to model, and therefore the modeling complexity is expected to be reduced further. Moreover, the conducted experiments provide evidence that general observations, like CPU usage and message frequency, are sufficient for reliable FDD estimates. In fact, the evaluation results are based exclusively on general system observations, while still providing reliable FDD estimations. Therefore, it seems that the necessary degree of inherent monitoring support of components is limited. In the evaluation, we compare our proposed FDD with two different reference approaches in respect to correctness and robustness. Due to a combination of high estimation quality and reduced modeling complexity, QBFD seems preferable for the robot domain.

REFERENCES

[1] D. Kirchner, S. Niemczyk, and K. Geihs, "RoSHA : A Multi-Robot Self-Healing Architecture," in *RoboCup2013: Robot World Cup XVII*. Eindhoven: Springer, 2013.

[2] V. Venkatasubramanian, R. Rengaswamy, and K. Yin, "A review of process fault detection and diagnosis Part I : Quantitative model-based methods," *Computers & Chemical Engineering*, vol. 27, 2003.

[3] D. Kirchner and K. Geihs, "Adaptive Model-based Monitoring for Robot Systems," in *International Conference on Intelligent Autonomous Systems*. Padova: IEEE, 2014.

[4] J. Hill and H. Sutherland, "OASIS : An Architecture for Dynamic Instrumentation of Enterprise Distributed Real-time and Embedded Systems," *Computer System Science and Engineering*, 2011.

[5] S. I. Roumeliotis, G. S. Sukhatmetand, and G. A. Bekey, "Sensor Fault Detection and Identification in a Mobile Robot," in *International Conference on Intelligent Robots and Systems*. IEEE, 1998.

[6] R. Golombek, S. Wrede, M. Hanheide, and M. Heckmann, "Online data-driven fault detection for robotic systems," in *IEEE International Conference on Intelligent Robots and Systems*, 2011, pp. 3011–3016.

[7] V. Venkatasubramanian, R. Rengaswamy, and S. N. Ka, "A review of process fault detection and diagnosis Part II : Qualitative models and search strategies," *Computers & Chemical Engineering*, vol. 27, 2003.

[8] V. Venkatasubramanian, R. Rengaswamy, and S. N. Kavuri, "A review of process fault detection and diagnosis Part III : Process history based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, 2003.

[9] G. Stanley, "A Guide to Fault Detection and Diagnosis," 2010. [Online]. Available: http://gregstanleyandassociates.com/whitepapers/ (accessed: 11.07.2014)

[10] G. Steinbauer, M. Martin, and F. Wotawa, "Real-Time Diagnosis and Repair of Faults of Robot Control Software," in *RoboCup 2005: Robot Soccer World Cup IX*, no. 1. Springer, 2006, pp. 13–23.

[11] R. Isermann, "Model-based fault-detection and diagnosis status and applications," *Annual Reviews in Control*, vol. 29, no. 1, 2005.

[12] W. E. Vesely, F. F. Goldberg, N. H. Roberts, and D. F. Haasl, "Fault tree handbook," DTIC Document, Tech. Rep., 1981.

[13] M. Ram Maurya, R. Rengaswamy, and V. Venkatasubramanian, "Application of signed digraphs-based analysis for fault diagnosis of chemical process flowsheets," *Engineering Applications of Artificial Intelligence*, vol. 17, no. 5, 2004.

[14] H. Boudali and J. Dugan, "A discrete-time Bayesian network reliability modeling and analysis framework," *Reliability Engineering & System Safety*, vol. 87, no. 3, 2005.

[15] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Pearson Education, 2003.

[16] P. O'Connor and A. Kleyner, *Practical Reliability Engineering*, 5th ed. John Wiley & Sons, Inc., 2012.

[17] D. Kirchner and D. Saur, "Reliable Robotics: Diagnostics++," in *ROS Developer Conference*, Stuttgart, 2013. [Online]. Available: http://roscon.ros.org/ (accessed: 11.07.2014)

[18] M. J. Druzdzel, "SMILE : A Development Environment for Graphical Decision-Theoretic Models," in *National Conference on Artificial Intelligence*, Orlando, 1999.

[19] J. Carlson, S. Member, and R. R. Murphy, "How UGVs Physically Fail in the Field," *Transactions on Robotics*, vol. 21, no. 3, 2005.

[20] J. Carlson and R. Murphy, "Reliability analysis of mobile robots," in *International Conference on Robotics and Automation*. IEEE, 2003.