A Method for learning a Fault Detection Model from Component Communication Data in Robotic Systems

Raphael Golombek, Sebastian Wrede, Marc Hanheide, Martin Heckmann

2010

Preprint:

This is an accepted article published in Proc. 7th IARP Workshop on Technical Challenges for Dependable Robots in Human Environments. The final authenticated version is available online at: https://doi.org/[DOI not available]

A Method for learning a Fault Detection Model from Component Communication Data in Robotic Systems

Raphael Golombek, Sebastian Wrede, Marc Hanheide, and Martin Heckmann

***Abstract*— A promising means to increase the dependability of a robotic system is to equip it with the ability to autonomously monitor it own system state and detect faults. In this contribution we propose a method for fault detection in robotic systems which exploits the concept of anomaly detection and learns a model based on dynamics in the system's internal exchange of data. Learning a model reduces the need for expert system-knowledge and enables on-line adaptation. Furthermore, communicated data as learning input enables the detection of subtle system failures such as resource starvation. The method in this contribution is applicable during runtime and can be used in an a-posteriori analysis of the system. The evaluate of the method takes place on a mobile robotic platform employed in human robot interaction scenarios.**

I. INTRODUCTION

Means to increase dependability have been identified throughout the whole life cycle of the system and can be roughly categorized into methods for prevention, tolerance, removal and forcasting of faults [1]. At least two of this categories require that the system is able to autonomously analyze its current state and detect faults during runtime. There exist several approaches for fault detection in autonomous robotic systems. In [6] model-based reasoning is used to detect faults in robot control software. Here, well understood reasoning algorithms can be used. But, defining rules for a complex robotic system strongly depends on the available expert knowledge. In [8] particle filters are used for real-time fault diagnosis in mobile robotic systems. The advantage of this approach is that it is able to cope with uncertainty. However, the scope of addressed faults is constrained to failures related to the locomotion of the robotic system.

Along these lines, a novel method for fault detection in robotics systems is presented. Following the concept of anomaly detection a probabilistic model is learned on internally exchanged data recorded during normal behavior of the system. This alleviates developers from the need to explicitly specify every exceptional or normal situation at design time of a to-be-controlled system. Furthermore, our approach abstracts from system specific characteristics before learning the probabilistic model and is therefore applicable to a wide range of systems.

R. Golombek and S. Wrede are with the Research Institute for Cognition and Robotics, Bielefeld University, P.O. Box 100131, Bielefeld, Germany `[rgolombe,swrede]@cor-lab.uni-bielefeld.de`

M. Hanheide is with the School of Computer Science, University of Birmingham, UK `m.hanheide@cs.bham.ac.uk`

Martin Heckmann is with the Honda Research Institute Europe, Offenbach, Germany `Martin.Heckmann@Honda-RI.de`

II. FAULT DETECTION METHOD

We consider a robotic system as a set of functional components communicating with each other to fulfill a given task thereby generating complex temporal communication patterns. We learn these patterns to build a model of normal behavior exploiting temporal correlations in the communication. To realize the fault detection existing approaches for anomaly detection could be used [2]. However, most of them do not consider temporal correlations in the communicated data which provide valuable information about the system behavior. Other assume the Markov property which is mostly invalid for the interaction in complex robotic systems. In the remainder of this section our method is described in detail.

a) Encoding Data: The components of a system generate domain specific output from different scales of measure (e.g., nominal, ordinal or a mixture of scales) resulting in a temporally ordered data sequence D. We decouple our model from implementation details of a specific system by applying a system specific mapping function $f : \mathcal{D}_j \mapsto \mathcal{E}$ to D. f maps each single data entity $d_j \subset D$ from its origin domain space $\mathcal{D}_j$ to a common domain space $\mathcal{E}$ resulting in a sequence E which we call an event sequence.

b) Learning The Model: We learn the model M based on an event sequence E recorded during normal behavior of the system. Traversing through E we calculate a probability distribution $P_{i,j}$ for each pair of unique events e_i, e_j. Thereby, $P_{i,j}$ models the timespan between the *last* occurrence of the event type e_i in E and the currently considered event e_j while traversing E. Consequently, the model M constitutes of the probability distributions $P_{i,j}$ for all possible combinations of two unique events e_i, e_j in E.

c) Evaluating The System State: This step corresponds to calculating a fitness value $\hat{s}_E$ for a sequence of events E which we call the *score* of E by averaging over the individual scores s_j of events e_j in E. Each s_j is calculated as the weighted sum over the probabilities $P_{i,j}(\Delta t)$ where Δt is the duration between the last occurrences of e_i and the current event e_j. The weight for a single $P_{i,j}(\Delta t)$ behaves inversely proportional to the entropy of $P_{i,j}$. A low entropy indicates low uncertainty in the distribution and high correlation between e_i and e_j.

d) Assessing The System State: Deciding whether the system behavior is normal or abnormal is done by comparing the score $\hat{s}_E$ of a sequence E against a threshold s^*. If $\hat{s}_E$ of E is high enough it is declared as normal. Otherwise abnormality is assumed. We determine the threshold s^* by calculating the *receiver operating characteristic(ROC)* *curve* [7] on test data and finding the optimal cut-off in the curve.

III. EVALUATION

The fault detection method is evaluated in the context of real human robot interaction scenarios. To measure the performance we induce several faults in the system and calculate the detection rate of our method. Additionally, we discuss the results compared to two base line methods, namely an entropy based and a Markov chain based approach.

As evaluation platform we use the mobile robot BIRON which is a robot companion equipped with social interaction capabilities [9], [4]. As a pool of interaction scenarios we chose the set of tasks defined for the robocup@home competition [5]. These tasks represent realistic human robot interaction scenarios and are fitted to evaluate the performance of the robot to navigate ("Follow Me" task), localize/recognize persons ("Who is Who" task) and interact in unknown environments ("Go Get it!" task).

We use the following set of failures for performance measurement. First we trigger the crash of a component called player which publishes laser data into the system. This crash results in the immobility of the robot. Second, we induce a fault that affects the slam component which generates hypothesis about the robot's position in the environment thereby degrading the navigation performance of the robot. By this means we test the fault detection rate on different levels of data processing i.e., barely processed sensory data and hypothesis of the robot's position. As third fault we trigger a resource starvation failure by inducing a high CPU-load in the system. Faults of this type are of interest as they occur often when integrating independently developed components into a system. Another fault occurs in the context of distributed systems and results from asynchronous clocks of the underlying operating systems. This fault affects all components which rely on synchronization based on the system clock. The last fault is an external disturbance of the speaker localization system. In noisy environments the localization mechanism is subject to an increased number of incorrectly detected speaking persons and consequently negatively affects the human robot interaction.

The evaluation of the fault detection method is work in progress. Up to know we evaluated the first four aforementioned faults during the "Follow Me" task with our fault detection method. The threshold calculated with the ROC-curve method results in $\hat{s}_E = 4.5$. We gather four data records of 100 second length while the system behaves normal as well as four records with the same durations for each fault. The faults are induced during normal behavior at specific point in time and lasts to the end of the record. Applying the method parametrized with the threshold $\hat{s}_E = 4.5$ on the records of normal behavior yields a False positive rate of $6,51\%$.

The first three cases were detected successfully. For the first fault reporting after detection was done with a True Positive rate(recall) of 1.00, and a precision value of 0.92. For the second fault the method yields a recall value of 0.99 and precision value of 0.94. In the case of the resource starvation fault the reached recall value was 0.98 and measuring the precision resulted in 0.93. All these values indicate a good performance for the first three faults. In the case of the asynchronous communication the fault is detected only sporadically. The reason behind this effect is that the asynchronicity becomes only apparent if components are effected that synchronize each other via the monitored communication channels which leads to discontinuous detection of this failure. This argumentation is supported by an a-posteriori analysis of the recorded data. This shows that during the detection of the fault an expected hypothesis about a recognized person is missing in the recorded data. This again can be ascribed to the asynchronous communication as too old hypotheses are discarded.

IV. CONCLUSION AND OUTLOOK

In this paper we introcude a novel method for fault detection in robotic systems learned on the basis of data interchange in the system. The method is largely independent from the specific system and shows promising results even for transient malfunctions in system behavior. Furthermore, it provides the basis for the developement of means to increase the dependability of autonomous robotic systems.

Next steps involve futher evaluation of the fault detection method in the remaining robocup tasks, comparison to the baseline methods and further a-posteriori analysis to demonstrate the practice of such an analysis in the developement cycle.

V. ACKNOWLEDGMENTS

The research project "An Autonomic Computing Approach for Systemic Self-Regulation" is supported by the Honda Research Institute Europe. [3]

REFERENCES

[1] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1:11–33, 2004.

[2] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 2009.

[3] Raphael Golombek et al. A probabilistic model for self-awareness in an event-based intelligent system. Technical report, Research institute for Cognition and Robotics (CoR-Lab), Bielefeld, Germany, 2009.

[4] Marc Hanheide, Sebastian Wrede, Christian Lang, and Gerhard Sagerer. Who am i talking with? a face memory for social robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, 2008. IEEE, IEEE.

[5] RoboCup league's committees. Robocup@home rules & regulations. 2009.

[6] G. Steinbauer, M. Morth, and F. Wotawa. Real-time diagnosis and repair of faults of robot control software. *Lecture Notes in Computer Science*, 4020:13, 2006.

[7] A.R. van Erkel and P.M.T. Pattynama. Receiver operating characteristic (roc) analysis: basic principles and applications in radiology. *European Journal of radiology*, 27:88–94, 1009.

[8] V. Verma, G. Gordon, R. Simmons, and S. Thrun. Particle filters for rover fault diagnosis. *IEEE Robotics & Automation Magazine special issue on Human Centered Robotics and Dependability*, 2004.

[9] Britta Wrede, Marcus Kleinehagenbrock, and Jannik Fritsch. Towards an integrated robotic system for interactive learning in a social context. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems - IROS 2006*, Bejing, 2007.