Preference-Based Fault Estimation in Autonomous Robots : Incompleteness and Meta-Diagnosis

Extended Abstract

Valentin Bouziat
Xavier Pucel
Stéphanie Roussel
ONERA
Toulouse, France
firstname.name@onera.fr

Louise Travé-Massuyès
LAAS-CNRS, University of Toulouse
Toulouse, France
louise@laas.fr

ABSTRACT

In autonomous systems, planning and decision making rely on the estimation of the system state across time. In this work, we use a preference model to implement a fault management strategy that selects a unique estimated state at each time point. If this strategy is not carefully designed, it can lead to incomplete estimators that meet a dead-end in some scenarios. Our goal is to detect such scenarios at design time and to be able to blame a subset of preferences causing them; those can be proposed to the designer for revision. To do so, we propose a method for detecting dead-end scenarios, introduce preference relaxation, and apply a consistency-based meta-diagnosis approach for identifying the sets of "faulty" preferences for a given dead-end scenario. We build upon SAT solvers for checking estimator incompleteness, and for consistency checking during meta-diagnosis.

KEYWORDS

Diagnosis and Abductive Reasoning ; Preference Modelling and Preference-Based Reasoning; Dependable Robots; SAT: Applications

ACM Reference Format:

Valentin Bouziat, Xavier Pucel, Stéphanie Roussel, and Louise Travé-Massuyès. 2019. Preference-Based Fault Estimation in Autonomous Robots : Incompleteness and Meta-Diagnosis. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 3 pages.

1 INTRODUCTION

Autonomous robots executing various tasks in a diversity of environments must handle unexpected aleas, both internal (failures, wear) and external (perturbations, environmental changes). Aleas impact the system's health, the goals that can be reached, and the way these goals can be reached.

Conformant and non-deterministic planning can cope to some extent with uncertainty on the state of the robot and its environment [4, 8, 11], and can be integrated with model-based diagnosis at execution time [15], or in self-healing plans [5]. In [12], probabilistic reasoning for decision is fully integrated and deployed.

This work assumes an approach that separates, on the one hand state estimation, and on the other hand decision making. We focus on state estimation and model the system as a partially observed finite state machine using boolean variables where both permanent and intermittent faults can be considered. We propose a preference-based state estimation approach that selects a unique estimate for the system state at each time step. Such an approach integrates well with decision functions that require totally defined inputs, which is the case of many task allocation algorithms for multi-robot systems [9] and many autonomous decision architectures [7].

2 BACKGROUND AND GOALS

In [13] and [2], the authors define an estimation framework composed of two parts: (1) a behavorial model (represented by logical constraints) that constrains the possible explanations for a given observed scenario, and (2) a fault management strategy (represented by a conditional preference model) that specifies which estimation is to be preferred, and under which conditions. Such a framework is used to implement rules of the form "if uncertain about fault f, and if condition x is met, act as if f was present/absent". Such rules can be combined to build elaborate fault management strategies. At each time step, the automatically generated estimator only keeps in memory the selected estimation, which makes this approach particularly scalable for complex robots, multi-robots systems or/and long-term autonomous missions.

However, in this approach the estimator can be undefined for some scenarios. More precisely, if it chooses an execution path different from that of the system, there may be observations for which it is unable to provide an estimation consistent with the previous estimation. Such situations are called *dead-end scenarios* and the estimator is said to be *incomplete*, *i.e.* it is not defined on all the system's possible observation sequences. [2] proposes a procedure that checks at design time whether the estimator is complete or has dead-ends. This procedure suffers from scalability and no insight is given on the causes of the existence of the dead-end scenarios, *i.e.* if it is due to system's dynamics, to poor observability, or to an inappropriate fault management strategy.

We assume that it is not possible to modify the behavioural model of the system but we allow preferences to be revised. We use the framework of [2] and our first contribution aims at the defining an efficient SAT-based procedure for finding dead-end scenarios of bounded length. Our second contribution targets the design of a meta-diagnosis procedure that identifies the minimal

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada. © 2019 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

sets of preferences in the fault management strategy that should be modified in order to eliminate a given dead-end scenario.

3 SYSTEM STATE ESTIMATION

We suppose that the system is governed by discrete dynamics where each time step lasts the same duration. System variables are partitioned into *observed* and *estimated* variables. A system state is an assignment of truth values to both observed and estimated variables and we assume that the initial system state is known. An assignment to observed variables is called an *observation.*

3.1 Behavioural model

The system state evolution is described as in symbolic model-checking [3] but we refer to the *previous* state instead of the *next* state. More precisely, we define two set of variables in order to represent the current and the previous time step. The behavioural model of the system is then a propositional formula over those two sets and is interpreted as a set of transitions in the classical way.

3.2 Preference model

State estimation is performed step by step as new observations are acquired from the system. Given an observation and a previous state, they may be many possible current states that we call candidates. In order to select only one estimation among the candidates at each time step, we follow the approach of [2] and use a preference model composed of a list of conditional preferences [16]. A conditional preference is a kind of "soft constraint" relating to one variable: it is only applied when there exists estimation candidates with both *true* and *false* values for the variable. Each preference favors a value for its associated variable according to the present observation and the previous state.

At each time step, the estimation problem consists in finding the unique valuation of unobserved variables that is consistent with the behavioural model, the current observation and the previous estimation, and that is optimal with respect to the conditional preferences.

4 ESTIMATOR COMPLETENESS

An estimator that produces an estimation sequence for any consistent observation sequence is said to be *complete*. It happens that some estimators are not complete, and the remaining of this presentation propose a method to test estimator completeness.

4.1 Dead-end scenarios

An observation can often be explained by several system states, and is associated to several estimation candidates. In this case, the estimator may select an estimation that differs from the system's actual state. In some cases, uncertainties may simply disappear as more observations are received from the system, and the estimator converges to the correct estimation. However, it can happen that an observation is inconsistent with the divergent estimation previously selected by the estimator. In this case, the set of estimation candidates is empty and the estimator cannot produce any output. The sequence of observations that led the estimator into it is called a *dead-end scenario.*

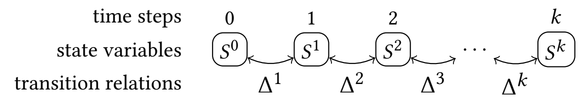

Figure 1: Unfolding transition relation Δ on k time steps.

4.2 Bounded dead-end detection

We unfold the formula of the behavioural model (denoted Δ) and for each time step we create a new set S of boolean variables as described in Figure 1. We use SAT model enumeration [10] to produce all the observation sequences consistent with the behavioural model of a fixed length k and we apply preferences with an existing estimator implementation based on MAX-SAT [13].

5 META-DIAGNOSIS

The problem of meta-diagnosis consists in finding which model elements are responsible for a particular outcome. In [1], several classes of elements are considered: inaccurate model of the system, altered observations, or errors in the reasoning process itself. In our approach, we use meta-diagnosis to find if a dead-end path can be caused by a poor fault management strategy. The behavioural model and observations are assumed to be always correct. Meta-diagnosis is an instance of a consistency based diagnosis problem [1, 6]: given a dead-end scenario and a subset of preferences, we want to know whether the estimator would be able to produce an estimation sequence if the given preferences were "relaxed".

5.1 Relaxed preference model

In order to investigate which preferences may have led to a dead-end, we relax preferences by omitting their constraints on the estimated state. While in conditional preference models, for a given previous state and observation, the preferred estimation is always unique. This is not the case for relaxed models : relaxing preferences produces additional estimates and estimation sequences.

5.2 Preference meta-diagnosis

Checking whether a relaxed preference model accepts some observation sequence can be done by a series of consistency checks. Searching for the smallest set(s) of preferences that eliminate a dead-end can be done with a classical consistency-based diagnosis algorithm [14]. To check whether a set of preferences is a meta-diagnosis, we perform a series of SAT queries that recreate the possible non-dominated estimation candidates at each time step. Our algorithms take as input the dead-end sequence, a tested preference, the behavioural model and the initial state.

6 CONCLUSION

This paper introduces a novel approach for blaming a fault management strategy. It follows a consistency-based meta-diagnosis strategy based on relaxation of conditional preferences and build upon SAT solvers. During our experiments, we noted that many dead-end scenarios reproduce the same pattern. A perspective is to identify such patterns for a more compact representation. Another perspective is to find relaxations that eliminate several dead-ends at once.

REFERENCES

[1] Nuno Belard, Yannick Pencolé, and Michel Combacau. 2011. A Theory of Meta-Diagnosis: Reasoning about Diagnostic Systems. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence IJCAI'11*. Barcelona, Spain, 731–737.

[2] Valentin Bouziat, Xavier Pucel, Stéphanie Roussel, and Louise Travé-Massuyès. 2018. Preferential discrete model-based diagnosis for intermittent and permanent faults. In *29th International Workshop on Principles of Diagnosis DX'18*. Warsaw, Poland. https://hal.archives-ouvertes.fr/hal-01796310

[3] Roberto Cavada, Alessandro Cimatti, Michele Dorigatti, Alberto Griggio, Alessandro Mariotti, Andrea Micheli, Sergio Mover, Marco Roveri, and Stefano Tonetta. 2014. The nuXmv Symbolic Model Checker. In *Proceedings of the International Conference on Computer Aided Verification CAV'14*. Springer, Vienna, Austria, 334–342. http://dx.doi.org/10.1007/978-3-319-08867-9_22

[4] Alessandro Cimatti and Marco Roveri. 2000. Conformant planning via symbolic model checking. *Journal of Artificial Intelligence Research* 13 (2000), 305–338.

[5] Alban Grastien, Enrico Scala, and Fondazione Bruno Kessler. 2017. Intelligent belief state sampling for conformant planning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. AAAI Press, 4317–4323.

[6] Walter Hamscher et al. 1992. *Readings in model-based diagnosis*. Morgan Kaufmann.

[7] Félix Ingrand and Malik Ghallab. 2017. Deliberation for autonomous robots: A survey. *Artificial Intelligence* 247 (2017), 10–44.

[8] Rune M Jensen, Manuela M Veloso, and Randal E Bryant. 2004. Fault Tolerant Planning: Toward Probabilistic Uncertainty Models in Symbolic Non-Deterministic Planning.. In *ICAPS*. 335–344.

[9] G Ayorkor Korsah, Anthony Stentz, and M Bernardine Dias. 2013. A comprehensive taxonomy for multi-robot task allocation. *The International Journal of Robotics Research* 32, 12 (2013), 1495–1512.

[10] António Morgado and Joao Marques-Silva. 2005. Good learning and implicit model enumeration. In *Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence ICTAI'05*. IEEE, Hong Kong, China, 6–pp.

[11] Christian J Muise, Vaishak Belle, and Sheila A McIlraith. 2014. Computing Contingent Plans via Fully Observable Non-Deterministic Planning.. In *AAAI*. 2322–2329.

[12] Sriram Narasimhan, Edward Balaban, Matthew Daigle, Indranil Roychoudhury, Adam Sweet, Jose Celaya, and Kai Goebel. 2012. Autonomous decision making for planetary rovers using diagnostic and prognostic information. *IFAC Proceedings Volumes* 45, 20 (2012), 289–294.

[13] Xavier Pucel and Stéphanie Roussel. 2017. Intermittent Fault Diagnosis as Discrete Signal Estimation: Trackability analysis. In *Proceedings of the 28th International Workshop on Principles of Diagnosis DX'17*. Kalpa Proceedings in Computing, Brescia, Italy.

[14] Raymond Reiter. 1987. A theory of diagnosis from first principles. *Artificial intelligence* 32, 1 (1987), 57–95.

[15] Gerald Steinbauer and Franz Wotawa. 2009. Robust Plan Execution Using Model-Based Reasoning. *Advanced Robotics* 23, 10 (2009), 1315–1326. https://doi.org/10.1163/156855309X462600

[16] Nic Wilson. 2011. Computational techniques for a simple theory of conditional preferences. *Artificial Intelligence* 175, 7-8 (2011), 1053–1091.