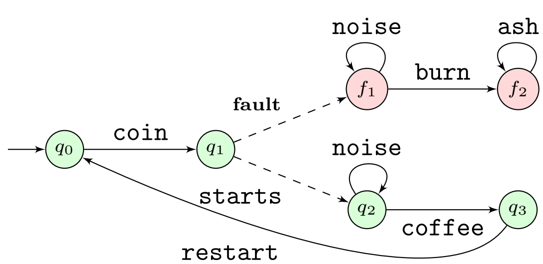
Active Prediction and Prediagnosability with Costs and Rewards

Serge Haddad[a], Engel Lefaucheux[b], Stefan Schwoon[a]

[a] *Université Paris-Saclay, LMF, ENS Paris-Saclay, CNRS Gif-sur-Yvette, France. {shaddad,schwoon}@lmf.cnrs.fr*
[b] *Université de Lorraine, CNRS, Inria, LORIA, France. engel.lefaucheux@inria.fr*

1. Context

A central task in a partially-observed system prone to errors is to detect or predict the occurrence of faults. A sequence of observations of such a system is said to be surely correct (respectively surely faulty) if all possible runs corresponding to this sequence are correct (respectively faulty); otherwise the observed sequence is ambiguous. While monitoring the system, a *diagnoser* should rule out ambiguities, and in particular detect that a fault occurred; the problem of existence of such a diagnoser is refered to as *diagnosability* [7]. In order to anticipate problems triggered by fault occurrences, one can also be interested in *predictors* that detect that a fault will eventually occur, and the *predictability* problem [3] is concerned with the existence of a predictor. *Prediagnosability* combines the benefits of predictability and diagnosability: depending on the observations, a *prediagnoser* behaves as a diagnoser or a predictor. Prediagnosability is of interest since predictability is more difficult to achieve than diagnosability, also prediagnosers can be seen as "as soon as possible" diagnosers [2].



As a quick example, consider the figure above abstracting a coffee machine. In a normal behaviour, you insert a coin in the machine, it starts correctly, makes some noise before giving you some coffee and resetting to its initial state. A faulty behaviour could occur during the starting phase however, leading to the machine burning. Moreover, the fault that leads to this behaviour is not observable to an external observer, a fact that is represented by using a dashed arrow. So, is this system diagnosable? In other words, can I identify every run which contains the fault? If the coffee machine starts burning, the fault can be detected — every run containing "burn" is surely faulty. However, the faulty sequence that loops infinitely on f_1, producing noises only cannot be distinguished from the correct sequence that loops on q_2. Thus there is a faulty run that remains ambiguous, and the system is therefore not diagnosable.

While these concepts were introduced for discrete-events, they have been extended into two complementary directions. On the one hand, when the system is controllable, one looks for a controller ensuring that all faulty runs will be eventually detected. Systems for which one can design a controller detecting the faults are called *actively diagnosable* [6, 4]. Similarly, when one can design a controller determining that a fault will occur whatever the future behaviour of the system, the system is called *actively predictable* [5]. On the other hand, when the discrete-event system is randomly driven, the requirement is slightly relaxed in at least two ways: (1) one looks for a diagnoser ensuring that all faulty runs will be almost surely eventually detected or (2) given any arbitrary $\varepsilon > 0$, one looks for a diagnoser ensuring that all faulty runs will be eventually detected with probability greater or equal than $1 - \varepsilon$ [8, 2].

In [1], these two directions have been combined for diagnosis where the authors study the decidability status of active diagnosability depending on the requirements for the active diagnoser in controllable probabilistic partially observable systems.

Very recently, the authors of this proposal have introduced *weights* for transitions in (non-deterministic or probabilistic) partially observed controllable system. Here, faulty runs are associated with *costs* or *rewards*. Thus the associated decision problems consist in determining the existence of an active diagnoser achieving (resp. not exceeding) some threshold for the minimal reward (resp. maximal cost).

Independently of the framework, there are two research directions for an internship. The semantic analysis consists in proposing several formalizations for an informal concept and comparing them. The algorithmic analysis consists in defining some corresponding decision problems and establishing their decidability status. In the positive case, one searches for tight lower and upper complexity bounds.

2. Location and goals

Depending on the conditions of the internship required by the institution of the intern, it can take place either in Paris-Saclay or in Nancy. The goals of this internship will depend on (1) the duration of the internship and (2) the theoretical background of the intern. These goals will be selected among the following ones:

- Proposing one or more specifications for the reward and cost of an active predictor or an active prediagnoser in non-deterministic transition systems;
- Proposing one or more specifications for the reward and cost of an active predictor or an active prediagnoser in probabilistic transition systems;
- Studying the corresponding decision problems in the non-deterministic and/or the probabilistic framework and, when decidable, establishing tight lower and upper complexity bounds for them.

References

[1] Bertrand, N., Fabre, E., Haar, S., Haddad, S., Hélouët, L.: Active diagnosis for probabilistic systems. In: Muscholl, A. (ed.) Foundations of Software Science and Computation Structures - 17th International Conference, FOSSACS 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings. Lecture Notes in Computer Science, vol. 8412, pp. 29–42. Springer (2014)

[2] Bertrand, N., Haddad, S., Lefaucheux, E.: Foundation of diagnosis and predictability in probabilistic systems. In: Raman, V., Suresh, S.P. (eds.) 34th International Conference on Foundation of Software Technology and Theoretical Computer Science, FSTTCS 2014, New Delhi, India, December 15-17, 2014. LIPIcs, vol. 29, pp. 417–429. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2014)

[3] Genc, S., Lafortune, S.: Predictability of event occurrences in partially-observed discrete-event systems. Autom. **45**(2), 301–311 (2009)

[4] Haar, S., Haddad, S., Melliti, T., Schwoon, S.: Optimal constructions for active diagnosis. J. Comput. Syst. Sci. **83**(1), 101–120 (2017)

[5] Haar, S., Haddad, S., Schwoon, S., Ye, L.: Active prediction for discrete event systems. In: Saxena, N., Simon, S. (eds.) 40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2020, BITS Pilani, K K Birla Goa Campus, Goa, India (Virtual Conference), December 14-18, 2020. LIPIcs, vol. 182, pp. 48:1–48:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2020)

[6] Sampath, M., Lafortune, S., Teneketzis, D.: Active diagnosis of discrete-event systems. IEEE Trans. Autom. Control. **43**(7), 908–929 (1998)

[7] Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., Teneketzis, D.: Diagnosability of discrete-event systems. IEEE Trans. Autom. Control. **40**(9), 1555–1575 (1995)

[8] Thorsley, D., Teneketzis, D.: Diagnosability of stochastic discrete-event systems. IEEE Trans. Autom. Control. **50**(4), 476–492 (2005)