Automatica 85 (2017) 468–477

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Decentralized observability of discrete event systems with synchronizations☆

Alessandro Giua [a,b], Cristian Mahulea [c], Carla Seatzu [b]

[a] Aix-Marseille University, CNRS, ENSAM, University of Toulon, LSIS UMR 7296, Marseille 13397, France
[b] Department of Electrical and Electronic Engineering, University of Cagliari, Piazza D'Armi, 09123 Cagliari, Italy
[c] Aragón Institute of Engineering Research (I3A), University of Zaragoza, Maria de Luna 1, 50018 Zaragoza, Spain

ARTICLE INFO

Article history:
Received 26 July 2016
Received in revised form 11 April 2017
Accepted 24 July 2017

Keywords:
Discrete event systems
Decentralized observability
Formal languages

ABSTRACT

This paper deals with the problem of decentralized observability of discrete event systems. We consider a set of sites each capable of observing a subset of the total event set. When a synchronization occurs, each site transmits its own observation to a coordinator that decides if the word observed belongs to a reference language K or not. Two different properties are studied: uniform q-observability and q-sync observability. It is proved that both properties are decidable for regular languages. Finally, under the assumption that languages K and L are regular, and all the events are observable by at least one site, we propose a procedure to determine the instants at which synchronization should occur to detect the occurrence of any word not in K, as soon as it occurs. The advantage of the proposed approach is that most of the burdensome computations can be moved off-line.

1. Introduction

Local observability is an important property of discrete event systems defined by Tripakis (2004b). The idea is the following: n local sites observe, through their own projection masks P_i (with $i = 1, \ldots, n$), a word w of symbols that is known to belong to a language L. A language $K \subset L$ is locally observable if, assuming all local sites send to a coordinator all observed words $P_i(w)$, the coordinator can decide for any w if the word belongs to K or to $L \setminus K$. Note that this property was shown in Tripakis (2004b) to be undecidable even when languages L and K are regular: this is due to the fact that the length of the observed words can be arbitrarily long and the information they contain cannot be compacted in a finite number of states. Moreover, for prefix-closed languages and three or more sites the problem is also undecidable (Tripakis, 2004b). On the contrary, assuming the observed words have bounded length q, one can define the property of *q-observability* that is decidable for arbitrary languages, since it must only be checked over a finite number of words. This property is closely related to local *diagnosability* as defined by Sampath, Sengupta, Lafortune, Sinnamohideen, and Teneketzis (1995). In fact, language K in this setting represents the set of all fault-free evolutions, while the larger set L also includes the faulty ones.

In this paper, which is an extended version of Cabasino, Giua, Mahulea, and Seatzu (2011), the considered problem is the following. Assume w describes the event driven evolution of a system. The coordinator can at any moment send a request to all local sites to get the locally observed words since the previous request: such a mechanism is called *synchronization*. After each synchronization (which in general is costly) a coordinator should be able to decide if, on the basis of the information received so far from the local sites, the word w generated belongs to the reference language K. We assume that the maximal number of events that can be generated by the system between two consecutive synchronizations is bounded. The coordinator should request as few synchronizations as needed to solve the observability problem. Also the distance between two consecutive synchronizations, expressed in terms of the number of events generated between them, may opportunistically vary with the word generated so far.

In this setting, although the basic notion of local observability given by Tripakis is still fundamental, two major extensions are needed. In fact the observability property defined in Tripakis (2004b) makes two rather restrictive assumptions.

- The first assumption is that the observability property is defined only with respect to words in L. On the contrary, in our

☆ This work has been partially supported by the Ministry of Economy and Competitiveness of Spain and FEDER under Grant number DPI2014-57252-R. The authors want to thank Maria Paola Cabasino, co-author of the conference paper on which is based this manuscript, for all her help during the initial version. The material in this paper was partially presented at the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), December 12–15, 2011, Orlando, Florida, USA. This paper was recommended for publication in revised form by Associate Editor Christoforos Hadjicostis under the direction of Editor Christos G. Cassandras.

E-mail addresses: alessandro.giua@univ-amu.fr, giua@diee.unica.it (A. Giua), cmahulea@unizar.es (C. Mahulea), seatzu@diee.unica.it (C. Seatzu).

http://dx.doi.org/10.1016/j.automatica.2017.08.009
0005-1098/

setting synchronizations occur repeatedly. Thus if a synchronization occurs after a word w has been generated we are interested in the observability of the residual language $w^{-1}K$, i.e., the set of words that can be generated after w, with respect to the residual language $w^{-1}L$. Correspondingly, we introduce the notion of *uniform observability*.

• The second assumption in Tripakis (2004b) is that when the observation starts, the word generated so far (that as discussed in the previous paragraph is always the empty word) is perfectly known. On the contrary, in our setting when a synchronization occurs the coordinator should be able to determine if the generated word belongs to the reference language K or not, but may not be able to unambiguously estimate it. Thus when next observation starts the word generated so far is only known to belong to a given set.

Combining the two extensions above, we introduce the notion of *q-sync observability*.

We point out a limitation of our approach: we assume that the coordinator at any time instant knows how many events have occurred so far, although it cannot directly observe *which* events have occurred. This assumption does not fit in a general asynchronous setting, where events may occur at arbitrary time instants. On the contrary, it makes sense in a synchronous setting where events occur with a fixed timing. Furthermore, we point out that our results can also be applied in those asynchronous cases in which any two consecutive events are spaced by a fixed known time interval. In such a case the coordinator knows an upper bound on the number of events that have occurred since last synchronization and can use this bound to determine when next synchronization should occur.

Literature review. Observability is a fundamental property that has received a lot of attention during the last decades. Several contributions have been presented in the framework of automata since late eighties and nineties (Caines, Greiner, & Wang, 1988; Caines & Wang, 1989; Kumar, Garg, & Markus, 1993; Özveren & Willsky, 1990). Caines et al. (1988) showed how it is possible to use the information contained in the past sequence of observations (given as a sequence of observation states and control inputs) to compute the set of consistent states, while in Caines and Wang (1989) the observer output is used to steer the state of the plant to a desired terminal state. A similar approach was used by Kumar et al. (1993) when defining observer based dynamic controllers in the framework of supervisory predicate control problems.

Özveren and Willsky (1990) proposed an approach for building observers that allows one to reconstruct the state of finite automata after a word of bounded length has been observed, showing that an observer may have an exponential number of states.

A very general approach for observability with communication has been presented by Barrett and Lafortune (2000) in the context of supervisory control, and several techniques for designing a possibly optimal communication policy have also been discussed therein. By optimal we mean that the local sites communicate as late as possible, only when strictly necessary to prevent the undesirable behavior. Our work is by large a special case of the architecture in Barrett and Lafortune (2000) because we allow communications only between the coordinator and the local observers – and not among local observers – and we do not consider a control problem but simply an observation one. There are, however, a few differences in our approach with respect to Barrett and Lafortune (2000) that motivate the need for additional investigation. First, we frame our results in the context of languages, rather than automata: this means that some of our definitions and results apply to possibly non regular languages. Secondly, while in Barrett and Lafortune (2000) communications are decided by the local observers and are triggered by the observation of an event, in our case the communications are triggered by the coordinator.

Preliminary results of this paper have been presented in Cabasino et al. (2011). The actual paper has been substantially improved by adding new theoretical results and new examples in order to clarify the theoretical results while the structure has been changed in order to improve the readability. Moreover, we are introducing a new notion called *uniform observability* that permits us to establish new connections between our work and the work of Tripakis (2004b).

Other interesting contributions related to the problem considered in this paper have been recently published. Fabre and Benveniste (2007) consider a distributed/modular system with several modules, each associated with a local observer/supervisor that only has access to the local observations and the model of the local module. Giua and Seatzu (2002) propose a procedure that produces an estimation of the state, while the special structure of Petri nets allows one to determine, using linear algebraic tools, if a given marking is consistent with the observed behavior without the explicit enumeration of the (possibly infinite) consistent set. Petri Nets with unobservable transitions, i.e., transitions labeled with the empty word, were studied in (Corona, Giua, & Seatzu, 2007). Here the notion of basis marking has been introduced. The idea is that under very general conditions, namely the acyclicity of the unobservable subnet, it is possible to characterize the set of markings consistent with an observation in terms of sequences of minimal length. The markings reached by these sequences are called basis markings and all other markings consistent with the observation can be obtained from the knowledge of this smaller set. Li and Hadjicostis (2007) consider the problem of state estimation in a Petri net framework assuming multiple observation sites with a partial order model of time. Finally, Hadjicostis and Seatzu (2016) focus on the problem of decentralized state estimation where two or more observation sites send information to a coordinator who aims to determine the set of possible current states of a given discrete event system modeled as a nondeterministic finite automaton.

Finally the approaches we present in this paper may also be useful to address other related problems in the area of discrete event systems, including (decentralized) diagnosis (Carvalho, Moreira, Basilio, & Lafortune, 2013; Yokota, Yamamoto, & Takai, 2016), prognosis (Takai, 2015; Yin & Li, 2016), and recovery, distributed supervisory control (Zhang, Cai, Gan, & Wonham, 2016) and minimal sensor activation for communicating observers (Sears & Rudie, 2016). Summarizing, the proposed results may be useful in all the applications where the state observation is done in a decentralized way, but it is important to minimize the cost and the energy consumption resulting from synchronization. A typical example in this context are sensor networks. Analogously, it may be important to minimize synchronizations in any application where security and privacy requirements are pressing, and when intrusions may suddenly occur.

Structure of the paper. The paper is structured as follows. In Section 2 we introduce basic notations on finite state automata and formal languages. In Section 3 we provide some language observability definitions and properties and discuss relationships among them. Section 4 focuses on uniform q-observability and provides specific results in the case of regular languages. A new property called q-sync observability is introduced and studied in Section 5. Again, special results are proved in the case of regular languages. The problem of determining the instants at which synchronize the observations from the different sites, so that a word not belonging to the reference language is identified as soon as occurred, is studied in Section 6. Conclusions are finally drawn in Section 7 where our future lines of research in this framework are pointed out.