



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

*Research Project for Smart Computing PhD Programme*

Model-based quantitative analysis  
for on-line diagnosis, prediction,  
scheduling and compliance  
evaluation in partially observable  
systems

**Applicant:** Tommaso PAPINI

## Introduction

More and more every year is seen all around the world an increasing use of *smart systems*, i.e. systems that incorporate the functions of sensing, actuation and control in order to describe and analyse a situation and make decisions based on the available data in a predictive and adaptive manner (i.e. performing *smart actions*).

Smart systems can differ much from one another, as they are used in many distinct fields. Smart systems range from sensor networks to smart buildings, from cyber-physical systems to smart cities, and also many different applications, like healthcare, security, entertainment or transportation [1, 2].

In this sense, *smart computing* is a multidisciplinary area that mainly aims to study how to design and build smart systems and how to use them in order to improve human life.

A notable example is the area of study of *Ambient Assisted Living* (AAL) which refers to the integration of electronic devices (made of sensors and actuators) into a habitation in order to improve the quality of everyday life by making mundane tasks easier [3, 4]. Special projects are being developed in order to create smart buildings for people with difficulties, such as people with handicaps or old people. These smart buildings consist of a series of sensors, through which an application can recognize which action the subject is performing, and some actuators, like a display that shows important information based on the data gathered, for example a reminder to take some pills at a certain time [5, 6].

Industrial Monitoring [7], which aim is to recognize faults and anomalies during the production process of an industrial plant in order to solve the problem as soon as possible, represents another application area for smart systems.. Further on, Industrial Monitoring is also very useful for predicting future faults, so to repair the faulty component even before the damage will happen, and scheduling the best time to perform maintenance, implementing for example a maintenance protocol.

## Goals

The main objective of this research project is to investigate model-based methods to perform diagnosis, prediction and action scheduling for real-time partially observable systems. This research project will especially focus on partially observable systems because most real systems and applications cannot be correctly modelled through fully observable systems, thus the need for partially observable systems. These systems, as the name suggests, are composed of a series of non-observable (or *hidden*) states and some other observable (or *visible*) states. As the system evolves from one hidden state to another it can also generate different observable events (also called *emissions*). The challenge here is to be able to perform quantitative analysis on the model of the system by taking into account just the observable data and, for example, trying to infer the non-observable part. On top of that, more complex models, such as mixed

stochastic/non-deterministic models, will be taken into account, in order to model and analyse more realistic systems.

The first problem that has to be solved is the *diagnosis* problem. The term diagnosis refers to the act of inferring the current system state taking into account only the observed events. In AAL, for example, diagnosis is crucial in order to understand, through only the data registered from the sensors, which action is the subject actually performing (also known as Activity Recognition, AR) [5]. In Industrial Monitoring, performing diagnosis would allow to detect a fault or an error before they produce a system failure, giving the chance to repair or patch the faulty component before irreparable damage is made, thus reducing maintenance costs.

A very interesting application of diagnosis is the evaluation of *compliance*. Compliance represent the likelihood that a certain path, or protocol, for the system is being correctly followed or applied. Real case examples are the two POR FESR 2014-2020 projects, namely the RACE and the DIGA projects. These two projects implements, respectively, a clinical and a maintenance protocol. For this two projects it has been suggested, as a future development, to include the evaluation of compliance in order to check whether a certain clinical or industrial protocol is being applied in a compliant way, in respect to the formal specification of the protocol, or not.

Once diagnosis for a system is done, it can be used to address a new problem, called *prediction*. As diagnosis means estimating the current state given present and past observations, prediction means estimating the evolution of the system's state taking into account the current diagnosis. In AAL, for example, prediction can be used to estimate the most probable future actions of the subject in order to display some advice/warning or start preparations for certain activities [6].

Both diagnosis and prediction are known to be achievable through the modeling of a partially observable system using a Hidden Markov Model (HMM). HMMs are statistical Markov models where the system being modelled is assumed to be a Discrete Time Markov Chain (DTMC) with hidden states that generates an observable state after each step of the process [8]. For this case, classical algorithms exist, such as the Viterbi algorithm, used for diagnosis to estimate the current system state. A viable alternative to HMM are Conditional Random Fields (CRF), which represent the discriminative counterpart of HMM and for which algorithms for diagnosis and prediction have already been studied [9]. If the underlying process is more complex than a DTMC, it might then be taken into consideration either a Continuous Time Markov Chain (CTMC), in case a continuous time representation is needed, or a Hidden non-Markov Model (HnMM), when the Markov condition (i.e. *memorylessness*) cannot be always guaranteed: for example the model can be a Semi Markov Process (SMP) [10] or a Markov Regenerative Process (MRP) [11]. These non-Markov type of processes are generally harder to analyse but allow to study a wider class of problems.

As prediction is performed on a certain system, it can be used as input to solve a third problem, called *scheduling*. Scheduling means calculating the best action to take and the best time (in the future) to take it, taking into account

the results of the prediction on the future states of system. In the context of AAL it could be some kind of message or warning being displayed, for example a remainder to take some pills, or suggestions related to the next activities that the subject is (most likely) going to perform. In the Industrial Monitoring context, action scheduling is crucial for different purposes, as in the case of maintenance interventions driven by early system failure estimation. Scheduling problems are usually modelled through the use of Markov Decision Processes (MDP) [12, 13].

A particularly interesting case of study is the one of on-line analysis. On-line analysis involves performing diagnosis, prediction and action scheduling as the process itself is evolving and changing state, as opposed to off-line analysis where all these actions are performed on a second time and not in parallel with the process. On-line analysis is crucial in many real case problem, such as AAL, where there is the need to recognize certain states of the system and take certain actions as the process evolves (and usually before a certain time limit).

## Approach

This research project is aimed to study the problems of on-line diagnosis, prediction and action scheduling for partially observable systems through model-based quantitative analysis. In particular, this research project will address the analysis of mixed models that combine both stochastic and non-deterministic elements and the evaluation of a compliance likelihood for such models in order to estimate whether a certain protocol, described as a series of time-stamped observable events, is being applied correctly or not.

Model-based quantitative analysis means that the goal is to first create a model of the system we want to analyse in order to be able to apply a series of algorithms for diagnosis and prediction, such as transient or steady-state evaluations. The model representing a certain system can be build either by hand, if the system is relatively simple, or through an automated process called *process mining* [14]. Process mining is composed by three main steps. The first step is called *process discovery* and it allows to generate the structure of the model (i.e. places, transitions, etc...) given a series of logged events. The next step is called *process enhancement* and is need to fill the model with the statistical data collected inside the logs (for example, transition durations, or probability density functions). The last step is known as *conformance checking* and is used to check if the model keeps being a good approximation of the system as new events are observed.

Another interesting concept that would be studied is that of *model@runtime* [15]. Models@runtime refers to a kind of models that are able to automatically adapt themselves, changing their structure and behaviour, in response to new observed events. This allow the model to be dynamic and adaptable with respect to the system being modelled. In [6] both process enhancement and the concept of models@runtime are implemented in order to achieve prediction and action scheduling in an AAL scenario.

Once a model of the system has been built, diagnosis, prediction and schedul-

ing can be achieved by performing both transient and steady-state analysis using well documented methods for markovian models, such as DTMC or CTMC, but especially for non-markovian ones, such as SMP or MRP [16, 17].

The actual quantitative analysis of models will make use of quantitative analysis tools, such as Oris [18].

Lastly, this research project is intended to cover both research and software development aspects of the problems presented. It is expected from this research project to produce new research and theoretical material but also to implement and test the new ideas presented.

## References

- [1] G. Akhras, “Smart materials and smart systems for the future,” *Canadian Military Journal*, vol. 1, no. 3, pp. 25–31, 2000.
- [2] Smartcomp. [Online]. Available: <http://www.smart-comp.org/>
- [3] V. Wadhawan, “Smart structures and materials,” *Resonance*, vol. 10, no. 11, pp. 27–41, 2005.
- [4] D. J. Cook, J. C. Augusto, and V. R. Jakkula, “Ambient intelligence: Technologies, applications, and opportunities,” *Pervasive and Mobile Computing*, vol. 5, no. 4, pp. 277–298, 2009.
- [5] L. Carnevali, C. Nugent, F. Patara, and E. Vicario, “A continuous-time model-based approach to activity recognition for ambient assisted living,” in *International Conference on Quantitative Evaluation of Systems*. Springer, 2015, pp. 38–53.
- [6] M. Biagi, L. Carnevali, M. Paolieri, F. Patara, and E. Vicario, “A stochastic model-based approach to online event prediction and response scheduling,” in *13th European Workshop on Performance Engineering*, October 2016.
- [7] L. H. Chiang, R. D. Braatz, and E. L. Russell, *Fault detection and diagnosis in industrial systems*. Springer Science & Business Media, 2001.
- [8] L. Rabiner and B. Juang, “An introduction to hidden Markov models,” *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.
- [9] J. Lafferty, A. McCallum, and F. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” in *Proceedings of the eighteenth international conference on machine learning, ICML*, vol. 1, 2001, pp. 282–289.
- [10] L. Lipsky, “Semi Markov Process,” in *Queueing Theory*. Springer, 2009, pp. 453–504.
- [11] D. Logothetis, K. S. Trivedi, and A. Puliafito, “Markov regenerative models,” in *Computer Performance and Dependability Symposium, 1995. Proceedings., International*. IEEE, 1995, pp. 134–142.
- [12] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.
- [13] C. Baier, H. Hermanns, J.-P. Katoen, and B. R. Haverkort, “Efficient computation of time-bounded reachability probabilities in uniform continuous-time Markov decision processes,” *Theoretical Computer Science*, vol. 345, no. 1, pp. 2–26, 2005.

- [14] W. Van der Aalst, T. Weijters, and L. Maruster, “[Workflow mining: Discovering process models from event logs](#),” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1128–1142, 2004.
- [15] G. Blair, N. Bencomo, and R. B. France, “[Models@ run. time](#),” *Computer*, vol. 42, no. 10, pp. 22–27, 2009.
- [16] A. Horváth, M. Paolieri, L. Ridi, and E. Vicario, “[Transient analysis of non-Markovian models using stochastic state classes](#),” *Performance Evaluation*, vol. 69, no. 7, pp. 315–335, 2012.
- [17] S. Martina, M. Paolieri, T. Papini, and E. Vicario, “[Performance Evaluation of Fischer’s Protocol through Steady-State Analysis of Markov Regenerative Processes](#),” in *Modelling, Analysis and Simulation of Computer and Telecommunication Systems*, September 2016.
- [18] G. Bucci, L. Carnevali, L. Ridi, and E. Vicario, “[Oris: a tool for modeling, verification and evaluation of real-time systems](#),” *International journal on software tools for technology transfer*, vol. 12, no. 5, pp. 391–403, 2010.