# Demo abstract of AVERIST: Algorithmic verifier of stability

Miriam García Soto[1]

***Abstract*— AVERIST[6] is a software tool which implements an algorithmic approach to verify stability of hybrid systems[5]. In particular, it analyzes stability of polyhedral switched systems. We illustrate the AVERIST performance through four easy examples, two stable and two unstable polyhedral switched systems.**

## I. Introduction

Biomedical control systems are hybrid systems. The development of such systems constitutes an emerging domain in areas as cardiovascular systems and endocrinology. One of the main features of such systems is a mixed discrete and continuous behaviour. Hybrid automaton is a formalism which captures such mixed evolution.

A fundamental property expected out of any control system is stability. The stability property assures that small perturbations in the input to the systems just result in small perturbations of the eventual behavior of the system.

The state of the art for stability verification relies on deductive methods. We propose an algorithmic approach.

## II. Architecture and design

AVERIST performs the following main functions:

- *State space partition*: determined by a finite set of linear expressions which divides the state space into regions.
- *Graph construction*: it constructs an abstract finite weighted graph which represents a simplification of the system. The nodes of the abstract graph correspond to the facets of the regions.
- *Graph analysis*: it model-checks the abstract system to either infer stability or to obtain a counterexample.

The software architecture is shown in Figure1. The Parma Polyhedra Library [4] is used to manipulate polyhedral sets, a reachability analyzer to determine the existence of edges and the GNU Linear Programming Kit GLPK [1] solver to compute the weights associated with such edges. All these steps are oriented to construct the abstract weighted graph. The graph is constructed and model-checked by using the NetworkX Python package [2]. All these utilities are included in the free open-source mathematics software system sage [3] and the tool is run in it.

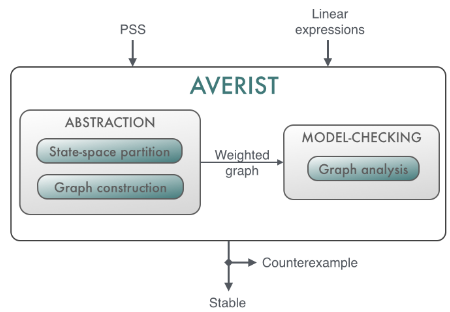


Fig. 1. AVERIST architecture

## III. Demo

We will show AVERIST performance on two stable and two unstable polyhedral switched systems as shown in Figure2. These examples are instantiated by using a markup language. The instances of the systems will be processed by AVERIST, which will generate output files containing relevant information about the run of the tool and the stability analysis. The obtained output includes the set of linear expressions used for the state space partition, the abstract weighted graph data, the original system elements and the time spent in the different processes of the tool.

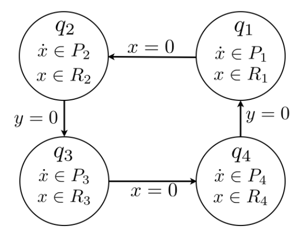


Fig. 2. Polyhedral switched system

## IV. Conclusion

The proposed demo shows an automatic approach for stability verification of hybrid systems. Input data can be easily defined by people with no experience in hybrid systems and running AVERIST does not require a formal knowledge on control systems and stability analysis.

## References

[1] Glpk: https://www.gnu.org/software/glpk/.
[2] NetworkX: https://networkx.github.io/.
[3] sage: http://www.sagemath.org/.
[4] R. Bagnara, P. M. Hill, and E. Zaffanella. The Parma Polyhedra Library: Toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems. *Science of Computer Programming*, 72(1–2):3–21, 2008.
[5] Pavithra Prabhakar and Miriam García Soto. An algorithmic approach to stability verification of polyhedral switched systems. In *ACC 2014*.
[6] Pavithra Prabhakar and Miriam García Soto. AVERIST: Algorithmic verifier of stability. In *NSV Workshop 2015*.

[1]Miriam García Soto is a research assistant at the IMDEA Software Institute, Madrid, Spain. Email: miriam.garcia@imdea.org