Equisolvability of Series vs. Controller's Topology in Synchronous Language Equations

Nina Yevtushenko[¶] Tiziano Villa[§] Robert K. Brayton[†] Alex Petrenko[‡]
Alberto L. Sangiovanni-Vincentelli[†]

[¶]Dept. of EECS
Tomsk State University
Tomsk, 634050, Russia

[§]PARADES
Via di S.Pantaleo, 66
00186 Roma, Italy

[†]Dept. of EECS
Univ. of California
Berkeley, CA 94720

[‡]CRIM
550 Sherbrooke West
Montreal, H3A 1B9, Can

Abstract

Given a plant M_A and a specification M_C, the largest solution of the FSM equation $M_X \bullet M_A \preceq M_C$ contains all possible discrete controllers M_X. Often we are interested in computing the complete solutions whose composition with the plant is exactly equivalent to the specification. Not every solution contained in the largest one satisfies such property, that holds instead for the complete solutions of the series topology. We study the relation between the solvability of an equation for the series topology and of the corresponding equation for the controller's topology. We establish that, if M_A is a deterministic FSM, then the FSM equation $M_X \bullet M_A \preceq M_C$ is solvable for the series topology with an unknown head component iff it is solvable for the controller's topology. Our proof is constructive, i.e., for a given solution M_B of the series topology it shows how to build a solution M_D of the controller's topology and viceversa.

1 Introduction

An important step in the design of complex systems is the decomposition of the system into a number of separate components which interact in some well-defined way. In this context, a typical question is how to design a component that combined with a known part of the system, called the context, conforms or satisfies or matches a given overall specification. This question arises in several applications ranging from logic synthesis to the design of discrete controllers.

In [3] we proposed a general frame based on defining equations over languages associated to the components of a given system. We introduced two composition operators for abstract languages: synchronous composition, $\bullet$, and parallel composition, $\diamond$, and we studied the most general solutions of the language equations $A \bullet X \subseteq C$ and $A \diamond X \subseteq C$ ($\subseteq$ denotes language containment), defining the language operators needed to express them. In particular we studied the solutions of the equations defined over finite state machines (FSMs) of the type $M_A \bullet M_X \subseteq M_C$ and $M_A \diamond M_X \subseteq M_C$, where M_A models the context, M_C models the specification and M_X is unknown. We refer to [4] for a report on FSM equations of the type $M_A \diamond M_X \subseteq M_C$ and to [2], Chap. 6, for a survey of previous work.

There are various FSM composition topologies of interest. For instance, the problem of designing a discrete controller that controls a given discrete plant in order to match a specification (represented by an FSM too) yields the so-called controller's (or supervisory control) topology (see [1] for an introduction to the discrete model matching problem). A more straightforward interconnection is the series topology where signals flow unidirectionally from a head FSM to a tail FSM. Figure 1 shows a series topology and a controller's topology. In the series topology M_X is the unknown head component, and M_A is the given tail component; in the controller's topology M_X is the unknown controller component, and M_A is the given plant component.

Given a plant M_A and a specification M_C, the largest solution of the FSM equation $M_X \bullet M_A \preceq M_C$ contains all possible discrete controllers M_X. Consider the constrained problem of practical interest $M_X \bullet M_A \cong M_C$, where both M_A and M_C are complete deterministic FSMs (DFSMs), and the objective is to find all complete DFSMs whose composition with M_A is equivalent to M_C ($\cong$ stands for equivalent). If M_A is not a Moore DFSM, the composition of a complete solution (when it is not a Moore DFSM either) with M_A may fail to pro-

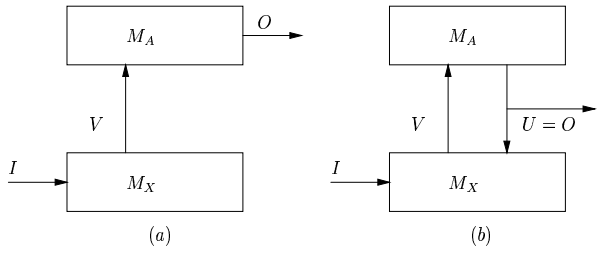

Figure 1. (a) 1-way cascade (or 1-way series) topology; (b) controller's (or supervisory control) topology.

duce a complete DFSM, i.e., the largest solution may contain complete DFSMs whose composition with M_A is not a complete DFSM (and so not be equivalent to M_C). Therefore if the goal is to select an 'optimal' controller, e.g., one with a minimum number of states, it will not be sufficient to find an 'optimal' reduction (contained FSM) of the largest solution, because this reduction may not be a solution of the original equation.

Instead the equation for the series topology has the property that every complete reduction of the largest solution is a DFSM whose composition with M_A yields a complete DFSM (due to the fact that if two FSMs are completely specified then their series composition also is completely specified). Can this fact be of help when solving (or modelling) the controller's topology ? To answer the question we should study the relation between the solvability of an equation for the series topology and of the corresponding equation for the controller's topology.

In this note we establish that, if M_A is a deterministic FSM, then the FSM equation $M_X \bullet M_A \preceq M_C$ is solvable for the series topology with an unknown head component iff it is solvable for the controller's topology. The non-trivial direction of the proof is going from the controller's topology to the series topology. The proof is based on the following fact: let M_D be a solution for the controller's topology (M_D has inputs I and O and output V), then M_C composed with M_D is a solution for the series topology ($M_C \bullet M_D$ has input I and output V). Indeed, when an input i is applied to such composition $M_C \bullet M_D$ M_C produces the reference output o; when this pair (i, o) is applied to M_D, by construction M_D produces the internal signal v under which M_A produces the reference output o generated by M_C under input i. The other direction relies on the fact that a solution for the series topology can be augmented with an inessential input o such that its next state and output functions do not depend on o.

Notice that the theorem of equisolvability can be proved for the general equation $M_X \bullet M_A \preceq M_C$, with no restriction on M_C (M_C is a NDFSM), whereas M_A should be a DFSM (or an appropriate restriction).

2 Conclusions

In this note we established that, if M_A is a deterministic FSM, then the FSM equation $M_X \bullet M_A \preceq M_C$ is solvable for the series topology with an unknown head component iff it is solvable for the controller's topology. Our proof is constructive, i.e., for a given solution M_B of the series topology it shows how to build a solution M_D of the controller's topology and viceversa.

A practical implication might be a procedure to compute discrete controllers by solving first a companion series topology equation and then transforming its solutions to solutions for the controller's topology. Notice that the largest solution for the series topology may have more states than the one for the controller's topology, however the latter has more inputs.

The first author was partly supported by the Russian Ministry of High Education. Both the first and the third author gratefully acknowledge the support of a NATO travel grant (NATO Linkage Grant No. 971217). The second author was partly supported by the MADESSII Project (Italian National Research Council). The fourth author gratefully acknowledges the support of NSERC (Grant OGP0194381).

References

[1] M. Di Benedetto, A. Sangiovanni-Vincentelli, and T. Villa. Model Matching for Finite State Machines. *IEEE Transactions on Automatic Control*, 46(11):1726–1743, December 2001.

[2] T. Kam, T. Villa, R. Brayton, and A. Sangiovanni-Vincentelli. *Synthesis of FSMs: functional optimization*. Kluwer Academic Publishers, 1997.

[3] N. Yevtushenko, T. Villa, R. Brayton, A. Petrenko, and A. Sangiovanni-Vincentelli. Sequential synthesis by language equation solving. In *The Proceedings of the International Workshop on Logic Synthesis*, June 2000.

[4] N. Yevtushenko, T. Villa, R. Brayton, A. Petrenko, and A. Sangiovanni-Vincentelli. Solution of parallel language equations for logic synthesis. In *The Proceedings of the International Conference on Computer-Aided Design*, pages 103–110, November 2001.