Compositional Modeling of Biological Systems in CospanSpan(Graph)*

Alessandro Gianola[1,2], Stefano Kasangian[3], Desiree Manicardi[4], Nicoletta Sabadini[4], and Simone Tini[4]

[1] Free University of Bozen-Bolzano, Italy
gianola@inf.unibz.it
[2] University of California San Diego (UCSD), USA
[3] Università degli Studi di Milano, Italy
[4] Università degli Studi dell'Insubria, Italy

Abstract. In this paper we investigate the expressiveness of the compositional formalism of CospanSpan(Graph) in order to model biological systems: first, we provide a compositional and timed description of the combined, complex system of the Heart and a Dual Chamber Pacemaker. Then, we consider as a case study the well-known gene regulation system in the Lac Operon in Escherichia coli.

Keywords: Automata · Compositionality · Categories · Open networks · Biological Systems

1 Introduction

The CospanSpan(Graph) model, introduced in [14,13], has been shown to model a variety of phenomena from asynchronous circuits to hierarchy, mobility and coordination [11]. The elements of the model are cospans and spans of graphs which here we shall call simply *Automata with interfaces*. Automata, since the seminal work of McCulloch and Pitts, have become the standard model for the specification and verification of *sequential* discrete dynamical systems. In recent years we have been assisting to a paradigmatic shift from sequential systems to *networks* of parallel, interacting components. Various models of automata with product (of states) have been proposed to represent *interactions* (Zielonka [21], Petri [17]). These models are rather natural, but unfortunately are not compositional, that is they lack a proper algebra. On the contrary, *compositionality*, i.e. the property of providing an algebraic calculus, is an essential feature of CospanSpan(Graph). In this approach, we provide explicitly operations that combine automata with interfaces and their connectors. Here, the operations can be interpreted in a very natural way as operations on automata with states and transitions, as well as interfaces and conditions. An expression in this algebra represents a hierarchical, reconfigurable *network of interacting components*.

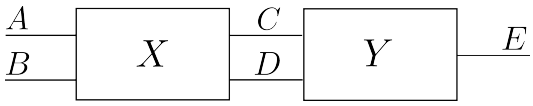
Automata Theory and Biology are very close disciplines, with a long tradition of reciprocal influences [3,15]. Many formalisms for modeling biological systems

have been proposed, mostly based on process algebras (e.g., [6,4]). In this paper we focus on the possibility an algebraic approach using CospanSpan(Graph) for the *compositional* description of biological systems. This is crucial for performing verifications tasks as well. Specifically, in [9] we gave a rather simple but compositional description of the heart. Here, we provide a complete description of a Dual Chamber Pacemaker following [12,1], but, for the first time, in a *compositional* way. So, a complete specification of the Heart-Pacemaker system can be provided. Finally, we consider as an additional case study the well-known gene regulation system in the Lac Operon of the Escherichia coli bacterium and we give a compositional description of it and its functioning.

2 CospanSpan(Graph): an algebraic formalism for automata networks

A full description of the algebra of CospanSpan(Graph) and its applications to reconfigurable networks of automata has been provided in a series of papers [14,13,19,11]. The algebra has a categorical flavor, since Span(C) and Cospan(C) were described in a general category C by Benabou in [2]. Here we recall that when C is the category of Graphs, the operations of the algebra correspond in a natural way to operations on automata with interfaces (and their connectors) that extend Kleene's algebras. Informally, a basic component is an automaton with states and transitions, i.e. a finite graph, plus: (i) a finite set of interfaces (i.e., communication ports); (ii) a selected subset of states, in analogy to initial and final states in classical automata; (iii) every transition has an effect, maybe ϵ, on *all* the interfaces. Hence, it is an *open* system, not input/output. Suitable operations could be defined on automata with interfaces [14,13]. The main operations are: (i) the tensor product, i.e. two automata in parallel without communication; (ii) the parallel with communication in which two automata can be joined on common interfaces and, for each pair of states, only the transitions that have the same effects on the common interfaces are possible; (iii) a sequential composition of automata through gluing selected states from both automata. Sequential and parallel feedback [11] can be derived from the full algebra.

In [14] an informal geometric description was introduced for these operations. For example, the parallel composition of two automata is pictured as:



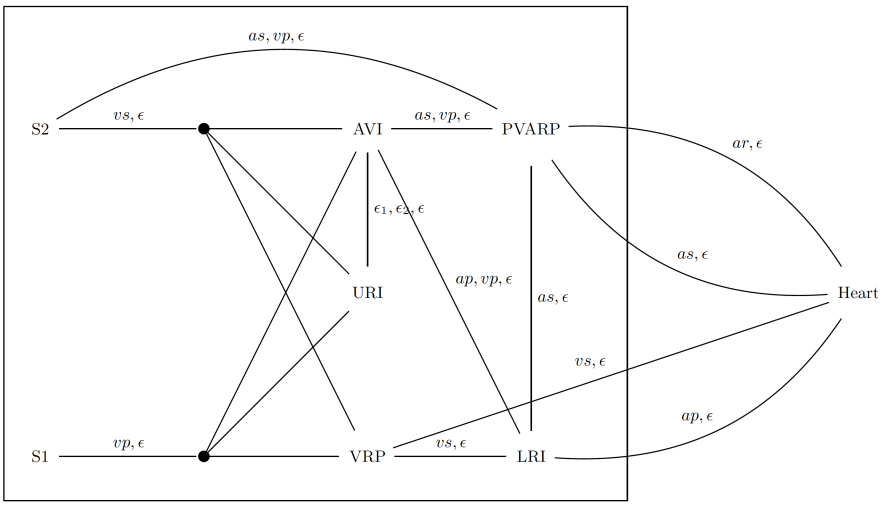
3 Pacemaker in CospanSpan(Graph)

In this section, through the use of timed Cospan-Span(Graph) [5], we provide the model of a pacemaker that communicates with the heart system described in [9]. A pacemaker is a system that promptly supplies electrical impulses to the heart in order to maintain an appropriate heart rate and also ventricular-atrial

synchrony. Different cardiac problems can occur, hence modern pacemakers are used in different ways: each of them has a different labeling. In particular, we model a Dual Chamber Pacemaker DDD − formalized in [12] using UPPAAL − that stimulates both the atrium and the ventricle.

In [12], the Pacemaker DDD is made up of five components: (i) **LRI** *Lower Rate Interval* (ii) **AVI** *Atrio-Ventricular Interval* (iii) **URI** *Upper Rate Interval* (iv) **PVARP** *Post Ventricular Atrial Refractory Period* and **PVAB** *Post Ventricular Atrial Blanking* (v) **VRP** *Ventricular Refractory Period.*

The next picture shows the pacemaker architecture and the communications with the heart system from [9]. Unlike [12], we add two components for broadcasting transmission: S1 and S2 - respectively for AP and VS - which transmits the signal to different other components simultaneously.



The pacemaker shown here was modeled considering the heart in brachycardia or with a regular beat − with 80 beats per minute. The constants TAVI, TLRI, TPVARP, TVRP, TURI and TPVAB − described in [12] − which control the duration of the operations, have the following values: (i) **TAVI**: 150 ms; (ii) **TLRI**: 1000 ms; (iii) **TPVARP**: 100 ms; (iv) **TVRP**: 150 ms; (v) **TURI**: 400 ms; (vi) **TPVAB**: 50ms.

We adopt the convention $Component$=$\{labels\}$, where labels are the proper labels of the interfaces and $Component$ corresponds to the automaton to connect.

We describe the **AVI** component (Figure 1), which maintains the appropriate interval between atrial and ventricular activation so it defines the longest interval between an atrial event and a ventricular event. If AVI does not detect any ventricular event (VS) after an atrial event (AS, AP), within TAVI, then AVI delivers a ventricular stimulation (VP). AVI has five interfaces: LRI = $\{ap, \epsilon\}$, PVARP = $\{as, \epsilon\}$, S2 = $\{vs, \epsilon\}$, URI = $\{\epsilon, \epsilon_1, \epsilon_2\}$ and S1 = $\{vp, \epsilon\}$. The transitions are:

$$
\begin{array}{cc}
ap, \epsilon, \epsilon/\epsilon, \epsilon : \text{-1} \rightarrow 0 & \epsilon, as, \epsilon/\epsilon, \epsilon : \text{-1} \rightarrow 0 \\
\epsilon, \epsilon, \epsilon/\epsilon_1, \epsilon : 0 \rightarrow 1 & \epsilon, \epsilon, \epsilon/\epsilon_2, \epsilon : 0 \rightarrow 1 \\
\epsilon, \epsilon, \epsilon/\epsilon_2, \epsilon : 1 \rightarrow 2 & \epsilon, \epsilon, \epsilon/\epsilon_1, \epsilon : 1 \rightarrow 2 \\
\ldots & \ldots \\
\epsilon, \epsilon, \epsilon/\epsilon_2, \epsilon : 149 \rightarrow 150 & \epsilon, \epsilon, \epsilon/\epsilon_1, \epsilon : 149 \rightarrow 150 \\
\epsilon, \epsilon, vs/\epsilon, \epsilon : 150 \rightarrow \text{-1} & \epsilon, \epsilon, \epsilon/\epsilon_1, \epsilon : 150 \rightarrow 151 \\
\epsilon, \epsilon, vs/\epsilon, \epsilon : 151 \rightarrow \text{-1} & \epsilon, \epsilon, \epsilon/\epsilon_2, \epsilon : 151 \rightarrow \text{-1} \\
\epsilon, \epsilon, \epsilon/\epsilon_1, \epsilon : 151 \rightarrow 151 & \epsilon, \epsilon, \epsilon/\epsilon, \epsilon : \text{-1} \rightarrow \text{-1}
\end{array}
$$

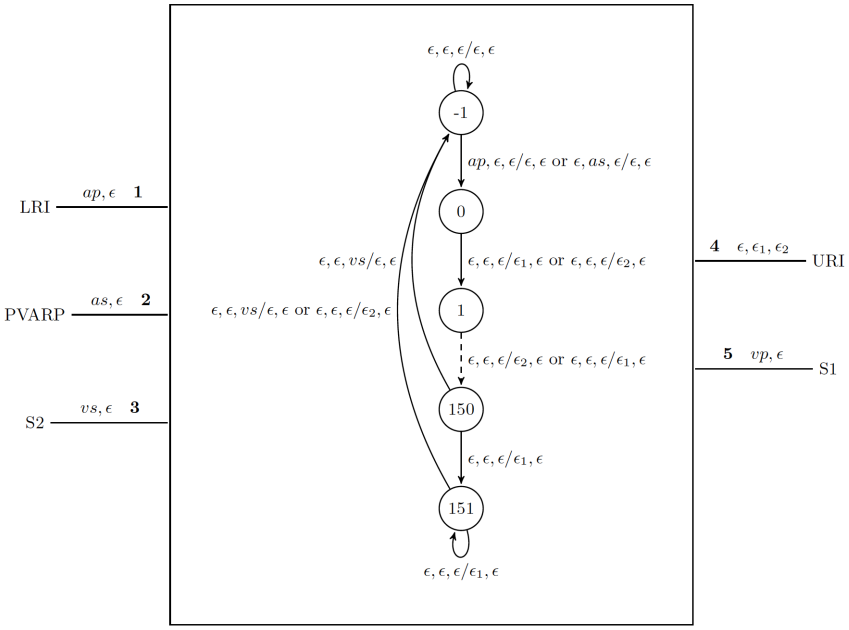


Fig. 1. AVI

The other components are described in detail in [10].

4 Lac Operon

In this section we formalize in CospanSpan(Graph) the Lactose Operon in the Escherichia coli bacterium, using for the first time a compositional framework.

The lactose operon in Escherichia coli is composed of a sequence of genes that are responsible for producing three enzymes for lactose degradation, namely the lactose permease, which is incorporated in the membrane of the bacterium and actively transports the sugar into the cell, the beta galactosidase, which splits lactose into glucose and galactose, and the transacetylase, whose role is marginal. The Lac Operon functionality depends on the integration of two different control mechanisms, one mediated by lactose and the other by glucose. The model, from

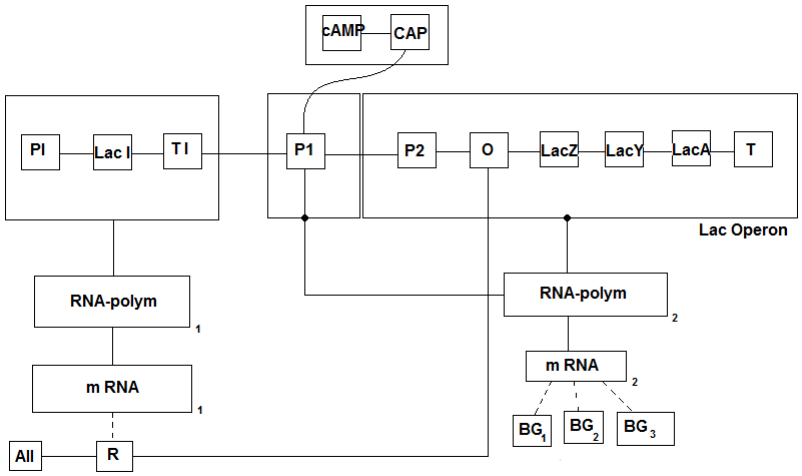


Fig. 2. Lac Operon model

[7,16,18], that we consider is depicted in graphical form in Figure 2. The DNA sequence of the Lac Operon (depicted in Figure 2) regulates the production of the enzymes, through the genes LacZ, LacY, LacA. The regulation process is as follows: gene LacI encodes the lac repressor R, which, in the absence of lactose, binds to gene O (the operator). Transcription of structural genes into mRNA is performed by the RNA polymerase enzyme, which usually binds to gene P2 (the promoter) and scans the operon from left to right by transcribing the three structural genes LacZ, LacY and LacA into a single mRNA fragment. When the lac repressor R is bound to gene O (that is, the complex R-O is present) it becomes an obstacle for the RNA polymerase, and transcription of the structural genes is not performed. On the other hand, when lactose is present inside the bacterium, it binds to the repressor thus inhibiting the binding of R to O. This inhibition allows the transcription of genes LacZ, LacY, LacA by the RNA polymerase. A second mechanism is relevant: when glucose is not present, the complex cAMP-CAP, which is present and acting on P1, can increase significantly the expression of lac genes. A complete description of the Lac Operon will be provided in a future paper, and it is interesting because the role of the Cospan structure is significant in the modeling.

5 Conclusions

In this paper we investigated the compositional feature of CospanSpan(Graph) in modelling biological systems. A compositional description of these systems is promising because it can provide effective verification techniques, using tools that have been developed for Span(Graph) [20]. Further developments could be, for example, a different type of pacemaker and the integration of time and probability [8] in the description of the Heart-Pacemaker System.

References

1. R. Alur and D. L. Dill. A theory of timed automata. *Theoretical Computer Science*, 1994.
2. J. Bènabou. Introduction to Bicategories. *Reports of the Midwest Category Seminar*, 47:1–77, 1967.
3. D. Besozzi, G. Mauri, G. Păun, and C. Zandron. Gemmating P systems: collapsing hierarchies. *Theor. Comput. Sci.*, 296(2):253–267, 2003.
4. L. Cardelli, E. Caron, P. Gardner, O. Kahramanogullari, and A. Phillips. A process model of actin polymerisation. *Electron. Notes Theor. Comput. Sci.*, 229(1):127–144, 2009.
5. A. Cherubini, N. Sabadini, and R. F. C. Walters. Timing in the Cospan-Span model. *Electr. Notes Theor. Comput. Sci.*, 104:81–97, 2004.
6. F. Ciocchetta and J. Hillston. Process algebras in systems biology. In *Proc. of SFM*, volume 5016 of *LNCS*, pages 265–312. Springer, 2008.
7. L. Corolli, C. Maj, F. Marini, D. Besozzi, and G. Mauri. An excursion in reaction systems: From computer science to biology. *Theoretical Computer Science*, 2012.
8. L. de Francesco Albasini, N. Sabadini, and R. F. C. Walters. The compositional construction of markov processes. *Applied Categorical Structures*, 19(1):425–437, 2011.
9. A. Gianola, S. Kasangian, D. Manicardi, N. Sabadini, F. Schiavio, and S. Tini. CospanSpan(graph): a compositional description of the heart system. *Fundamenta Informaticae*, 171 (1-4):221–237, 2020.
10. A. Gianola, S. Kasangian, D. Manicardi, N. Sabadini, and S. Tini. Compositional Modeling of Biological Systems in CospanSpan(Graph) (Extended Version). Technical report, `https://gianola.people.unibz.it`, 2020.
11. A. Gianola, S. Kasangian, and N. Sabadini. Cospan/Span(Graph): an algebra for open, reconfigurable automata networks. In *Proc. of CALCO*, pages 2:1–2:17, 2017.
12. Z. Jiang, M. Pajic, S. Moarref, R. Alur, and R. Mangharam. Modeling and verification of a dual chamber implantable pacemaker. In *Proc. of TACAS*, pages 188–203, 2012.
13. P. Katis, N. Sabadini, and R. Walters. A formalization of the IWIM model. In *Proc. of Coordination Languages and Models*, volume 1906 of *LNCS*, pages 267–283. Springer, 2000.
14. P. Katis, N. Sabadini, and R. F. C. Walters. Span(Graph): A categorial algebra of transition systems. In *Proc. of AMAST*, pages 307–321, 1997.
15. C. Martín-Vide, G.Mauri, G. Paun, G. Rozenberg, and A. Salomaa, editors. *Membrane Computing, International Workshop, WMC 2003, Tarragona, Spain, July 17-22, 2003, Revised Papers*, volume 2933 of *Lecture Notes in Computer Science*. Springer, 2004.
16. G. Pardini, R. Barbuti, A. Maggiolo-Schettini, P. Milazzo, and S. Tini. Compositional semantics and behavioural equivalences for reaction systems with restriction. *Theoretical Computer Science*, 2014.
17. C. A. Petri and W. Reisig. Petri net. *Scholarpedia*, 3(4):6477, 2008.
18. F. J. Romero-Campero and M. J. Pérez-Jiménez. Modelling gene expression control using P systems: The lac operon, a case study. *Biosyst.*, 2008.
19. N. Sabadini, F. Schiavio, and R. Walters. On the geometry and algebra of networks with state. *Theor. Comput. Sci.*, 64:144–163, 2017.
20. F. Schiavio. SpanTools. `https://sourceforge.net/projects/spantool/`.
21. W. Zielonka. Notes on finite asynchronous automata. *ITA*, 21(2):99–135, 1987.