

Formal Verification for Natural and Engineered Biological Systems

Hillel Kugler

Faculty of Engineering, Bar-Ilan University

Ramat Gan, Israel

hillelk@biu.ac.il

Abstract—Computational modeling is now used effectively to complement experimental work in biology, allowing to identify gaps in our understanding of the biological systems studied, and to predict system behavior based on a mechanistic model. We provide an overview of several areas in biology for which formal verification has been successfully used. We highlight examples from both natural and engineered biological systems. In natural biological systems the main goal is to understand how a system works and predict its behavior, whereas for engineered biological systems the main goal is to engineer biological systems for new purposes, e.g. for building biology-based computational devices. We compare between the challenges in applying formal verification in biology and the application to traditional domains. Finally, we outline future research directions and opportunities for formal verification experts to contribute to the field.

Index Terms—Formal Verification, Computational Systems Biology, DNA Computing, Biological Systems Modeling

I. INTRODUCTION

We provide an overview of the use of formal verification methods in biology and outline future research challenges and opportunities. We broadly divide applications of formal verification methods in biology to natural biological systems and engineered biological systems.

II. NATURAL BIOLOGICAL SYSTEMS

Biologists employ a wide range of experimental and theoretical approaches to decipher the mechanisms underlying the dynamic behavior of biological systems. Biological systems are the ultimate reactive system [4] as they need to constantly react to changes in their environment to maintain life. Viewing biological systems as reactive systems leads naturally to specifying their behavior using a subset of temporal logic [1], [2] and applying formal verification to prove system properties. An important class of computational models deals with the genotype/phenotype question - how does the genetic information in a cell or organism determine the dynamics and observable behavior. In particular, Gene Regulatory Networks (GRNs) capture the interactions between the genetic components of the system and define the system dynamics. Simulating GRN models allows us to study the network behavior under different experimental conditions including genetic mutations. Formal verification becomes useful since

The research was supported by the Horizon 2020 research and innovation programme for the Bio4Comp project under grant agreement number 732482 and by the ISRAEL SCIENCE FOUNDATION (grantNo. 190/19).

it is important to verify that known experimental results are indeed satisfied by the GRN models, and to identify behavior that may lead to new biological insights. In addition to using formal verification, synthesis methods [3], [7] can automate the process of deriving consistent models and offers a way to synthesize all solutions of a certain class and thus avoid the inherent bias in focusing on one model.

III. ENGINEERED BIOLOGICAL SYSTEMS

Engineering biological devices to perform specified computation is of major interest due to the potential of utilizing the inherent parallelism in biological components to speed computation, construct low energy consuming devices and interface with biological material. We focus on two main approaches of engineering such systems: DNA Strand Displacement Systems (DSD) [6] that utilize the complementarity of DNA sequences to bind together and perform designed reactions, and Network-Based Biocomputation (NBC) [5] that uses biological agents that operate in parallel to explore manufactured planar devices. We show computational design tools for these approaches and the use of formal verification methods and tools to improve the development process and ensure robust system design.

REFERENCES

- [1] Chabrier, N., Fages, F.: Symbolic model checking of biochemical networks. In: Proceedings of the First Intl. Workshop on Computational Methods in Systems Biology. pp. 149–162. Lecture Notes in Computer Science, Springer (2003)
- [2] Fisman, D., Kugler, H.: Temporal reasoning on incomplete paths. In: 8th International Symposium On Leveraging Applications of Formal Methods, Verification and Validation (ISOLA'18). Lecture Notes in Computer Science, vol. 11245, pp. 28–52. Springer (2018)
- [3] Goldfeder, J., Kugler, H.: Temporal logic based synthesis of experimentally constrained interaction networks. In: MLCSB'18. Lecture Notes in Computer Science, vol. 11415, pp. 89–104 (2019)
- [4] Harel, D., Pnueli, A.: On the development of reactive systems. In: Apt, K.R. (ed.) Logics and Models of Concurrent Systems. NATO ASI Series, vol. F-13, pp. 477–498. Springer, New York (1985)
- [5] Nicolau, D.V., Lard, M., Korten, T., van Delft, F.C., Persson, M., Bengtsson, E., Månsson, A., Diez, S., Linke, H., Nicolau, D.V.: Parallel computation with molecular-motor-propelled agents in nanofabricated networks. Proceedings of the National Academy of Sciences **113**(10), 2591–2596 (2016)
- [6] Qian, L., Winfree, E.: Scaling Up Digital Circuit Computation with DNA Strand Displacement Cascades. Science **332**(6034), 1196–1201 (2011)
- [7] Yordanov, B., Dunn, S.J., Kugler, H., Smith, A., Martello, G., Emmott, S.: A method to identify and analyze biological programs through automated reasoning. npj Systems Biology and Applications **2**(16010) (2016)