Two Views of a Biology-Computer Science Alliance [⋆]

Irun Cohen[1] and David Harel[2]

[1] Department of Immunology
The Weizmann Institute of Science, Israel
`irun.cohen@weizmann.ac.il`
[2] Department of Computer Science and Applied Mathematics
The Weizmann Institute of Science, Israel
`dharel@weizmann.ac.il`

Abstract. The editors of these *CoSMoS* Workshop Proceedings have invited us to describe the two sides of our joint approach to computer simulation of biological systems: the biological side and the computer science side. Irun Cohen, an immunologist, will voice the biological side; David Harel, a computer scientist, will voice the computer science side.

1 Cohen

I was led to David Harel and to the tools and thinking of computer science by a combination of factors emerging from my research into autoimmune diseases and the regulation of the immune system. Some of these factors are common to biology research generally and some are linked specifically to immunology and to my particular way of thinking. I shall list the factors that appear to have been the most influential for me, beginning with the general and proceeding to the particular.

1.1 The mass of experimental information staggers the imagination.

A biologist who zooms out for a comprehensive view of his or her field of study is frustrated; the mass of data defies memory and intuitive understanding. It appeared clear to me that only a computer could remember all the details and might help the biologist sort them out.

1.2 Pleiotropism and redundancy thwart understanding.

Human understanding is most comfortable with one-to-one, linear causality; for example, intuition early on posited that each DNA gene sequence is expressed

---

[⋆] Part of this work was supported by The John von Neumann Minerva Center for the Development of Reactive Systems at the Weizmann Institute of Science and by an Advanced Research Grant from the European Research Council (ERC) under the European Community's 7th Framework Programme.

through one messenger RNA and translated into one protein with a single function. Now we have learned about DNA methylation and other epigenetic control mechanisms; alternative splicing; post-transcriptional and post-translational modifications; single proteins with five or ten different functions. In my own field, we see that a single cytokine can make some cells die, some cells live and grow, some cells differentiate – depending on the type and states of the cells. Redundancy and pleiotropism – multi-functionality – are the norm [5]. Most human minds find it difficult to keep track of pleiotropic and redundant systems; our thinking favors simple engineering diagrams. We need the help of systems and design experts to handle all the redundant and multi-functional details of living organisms.

1.3 Experimental analysis is static; living systems are dynamic.

Living systems are dynamic at every level – molecules, cells, organisms, populations; continuous interaction is the essence of biology. Yet, the experimental method almost always involves a controlled but static analysis of the defining dynamics. If we want to see how living systems work, we must go beyond analysis and study running simulations that can integrate the piecemeal experimental data to recreate a working, dynamic whole.

1.4 We will not be able to reduce living systems to fundamental laws of nature; biology is not like physics.

The paradigm established by physics asserts that truly scientific understanding is founded on simple underlying laws, preferably quantitative, that account for the manifest complexity of the real world. Science, so viewed, is the quest for fundamental laws of nature. The laws of nature endure; such laws define being. The transient, mortal details of life are trivial accidents. Physics is interested in being; physics would see no fundamental difference between C. elegans and H. sapiens; in essence, they both operate according to the same chemical reactions and cellular networks, albeit to different degrees of complexity. Biology, in contrast to physics, strives to understand the impermanent differences between C. elegans and H. sapiens; the differences are based on cursory details; the differences are not being, but becoming. Biology cannot reduce the messy details of the living organism to explanatory, fundamental laws of nature; biology is defined by attending to the fleeting details. Biology is interested in the reactive designs of life. Biologists, like me, need the help of computer scientists and their computers to record, characterize, and catalog the complex details of living systems, and to render them quantitative and dynamic. Computer scientists know about design principles. Indeed, the only principle of biology that approaches the status of a law is the notion of evolution through genetic variation and survival of the fittest – evolution is a design for becoming, not a law of being.

1.5 Mathematical analysis has been incomplete.

Biologists have collaborated with mathematicians to clarify biologic complexity by reducing it to relatively simple abstract relationships that can be analyzed by differential equations and other mathematical tools. My formative experience with mathematical theoreticians has been fruitful [14, 27, 2, 24–26, 10] and has prepared me for later work with Harel and computer science. But these forays into theory could not take into account the rich details of the actual biology behind the phenomena that were modeled mathematically. The math misses the ontogeny.

1.6 The immune system is cognitive and computes the state of the body.

The prevailing view has been that the immune system has only one particular bias: it must not respond to self-molecules; the functional immune repertoire is selected by the foreign antigens that happen to enter the body. In contrast, my experimental investigations of autoimmunity and autoimmune diseases led me to conclude the immune system is cognitive – it has a built-in representation of particular self-molecules and molecular contexts that guides its response decisions both to self and to foreign; the immune system, as it where, knows what it is looking for [11, 3, 4].

The immune system, moreover, is engaged in an ongoing computation of the state of the body: Its receptors gather information about the body (injury, infection, the need to grow blood vessels or destroy them, the need to stimulate cells or kill them, the need to heal wounds, etc) and tranduces this input information into an output of regulated inflammation that feeds back to modify the body and the immune system itself [5–7]. The details are beyond the scope of this chapter. The point here is that my appreciation of the internal structure, the cognitive behavior and computational function of the immune system led me to see what David Harel and computer science might have to offer.

2 Harel

2.1 Origins

I had been working since the early 1980's on the problem of specifying the behavior of complex *reactive systems* [21]. This started via a one-day-per-week consultation job at the Israel Aircraft Industries (IAI), working with the avionics team of a fighter aircraft project. During that period, I proposed a *visual formalism* [16] — a rigorous diagrammatic language — for specifying reactive behavior, which I called *statecharts* [15]. This language has been around now for exactly 25 years and is considered a useful way of specifying behavior of such complicated systems [20]. It is in broad use in many industries, from cellular telephones to aircraft, automobiles, interactive software systems, and the like.

During that period the two main issues I was obsessed with were specifying the system's behavior in ways that would be intuitive and natural, but, at the very same time, formal and executable. We want a medium with which it is easy to specify and model and which is easy to understand, but, on the other hand, it must also have rigorous mathematical underpinnings, detailed and precise enough so that the system's full behavior can be executed (simulated) and analyzed by computer. The language of statecharts is a modest attempt to approach these two goals, in that it is highly diagrammatic, and seems to fit the way people think about behavior, yet it is as rigorous as any mathematical equation or computer program.

2.2 Biology and computer science

In the late 1980's it occurred to me that such problems of specifying behavior are not only common and problematic in human-made systems, such as the ones mentioned above, but might also be problematic in modeling Nature — in particular, biological systems. It seemed that the notion of a reactive system and the great difficulties that lie in designing them, are common to biological systems too. Perhaps the methods and languages used to design human-made ones could be used to help understand biology by "reverse engineering" it. However, having almost no experience in biology I wasn't able to take this much further on my own. And then, as Irun Cohen wrote in his portion of this paper, we happened upon each other, which led us to carry out an initial modest project with Na'aman Kam, a joint student who indeed started to use statecharts to specify biological phenomena [22].

The current paper will not be discussing the details of the work that we have done together in the last decade or so (see, e.g., [12, 13, 28, 29], but, in retrospect, I would submit that modeling, specifying and analyzing biology is one of the most exciting imaginable uses of computer science. This includes a lot of work by systems biologists, by bioinformatics researchers, but also by people like us who are doing something that is not exactly bioinformatics and is not exactly systems biology either. Rather, it is an attempt to mimic and imitate biology using means taken from software engineering and computer science, in an attempt to understand an entire system in a broad way [18, 19]. Many people, including myself, believe that computer science — as opposed to computer software or computation per se — will play a role in the science of the 21st century (which will no doubt be dominated by the life sciences) similar to the role played in the science of the 20th century (which was dominated by the physical sciences).

2.3 On aircraft and elephants

If we were to take as a typical example of a complex human-made system an F-16 or F-35, we might want to take as an example a complex biological system a multi-cellular organism, such as an elephant, or a lizard, or a fly, or a worm. The former has to be specified, designed, built, tested, debugged, maintained, and so on, and the latter has to be understood, which is by no means a small feat! In fact, the work that Irun and I have been doing with a cadre of extremely talented students raises the possibility of carrying out a *whole organism project* (WOP), the main goal of which would be to model an entire multi-cellular organism [17]. The organism would be modeled up to a certain pre-agreed-upon level of detail, and we want a model that would be fully and interactively executable. We would be reverse engineering an elephant, so to speak, rather than engineering an F-35. . .

The benefits of a successful WOP are almost unimaginable. Imagine us being able to actually simulate an elephant, in its entirety, have it develop from a fetus, have it walk and eat and play, and produce offspring.Moreover, and perhaps most importantly, during all of that the user of the model can intervene, by anything from switching off a gene, to treading on the elephant's foot and seeing what happens. If we could do that, especially if the model would reach down into the cellular and sub-cellular levels, there's almost no limit to the kinds of things we might be able to discover.

My claim is that although such a WOP is an almost unimaginably complicated task, even for a small organism, and one would need something like 15-20 years to approach it, it is possible. An elephant is obviously not a particularly good idea, but something smaller and more manageable is, such as the *C. elegans* nematode, which has a little over 1000 cells, all easily traceable in the lab. This is a multi-cellular organism of sufficient complexity to teach us an enormous amount about life, sickness and death, and the processes leading to them, but it is also of sufficiently modest proportions so as to be imaginable in a long, but reasonable, time frame [17, 19].

2.4 Visuality and formality

It is worth re-emphasizing that the recommendation is that modeling biology would have to be done using approaches that are intuitive enough so that biologists, who are not trained as mathematicians or as computer scientists, would be able to do much of the modeling themselves. My feeling is that the education that biologists will be getting in the next 10-15 years will bring them closer to the ability to master techniques for the modeling and analysis of complex systems, just as the education physicists got many tens of years ago, became a lot more mathematical, enabling them master and use mathematical techniques to model and analyze physical phenomena. Of course, as mentioned earlier, the languages and approaches used should be intuitive, but formal, so they their attractiveness to the biological community notwithstanding, they would be rigorous enough to be fully executed.

These days, biologists use many kinds of diagrams to describe pathways and networks, and the interdependency of genes and other intra-cellular entities. Many of these diagrams are informative and clear, but for the most part they are not sufficiently formal. Most of the approaches that advocate such diagrams do not give rise to full execution. First of all, this is due to lack of sufficiently detailed biological data and parameters; but more importantly, it is because the semantics of the diagrams themselves does not lend itself to full executability; see

[23]. In contrast, the approaches we have been using (for example, statecharts-based modeling) are such that once you have your model intact, it can be run, or executed just like any other computer program.

2.5 One more word on the collaboration

From my personal point of view, the collaboration with Irun, which started in the late 1990's, has been one of the most productive and exciting periods of my scientific life. I hope that we will continue to work for many years and that our work will be at least as exciting as it has already been. Perhaps we will indeed be able to contribute a little to the feeling that the marriage between computer science and the life sciences is already, but will become even more so, one of the most fruitful marriages in modern science.

References

1. H. Amir-Kroll, A. Sadot, I.R. Cohen and D. Harel. GemCell: A Generic Platform for Modeling Multi-Cellular Biological Systems. *Theoret. Comput. Sci.* 2008; **391**(3):276-290.
2. H. Atlan and Cohen IR. Immune information, self-organization and meaning. *Int Immunol* 1998; **10**(6):711-7.
3. I.R. Cohen. The cognitive principle challenges clonal selection. *Immunol Today* 1992; **13**(11):441-4.
4. I.R. Cohen. The cognitive paradigm and the immunological homunculus. *Immunol Today* 1992; **13**(12):490-4.
5. I.R. Cohen. *Tending Adam's Garden: Evolving the Cognitive Immune Self.* Academic Press, London, UK, 2000.
6. I.R. Cohen. Discrimination and dialogue in the immune system. *Semin Immunol* 2000; **12**(3):215-9; 321-323.
7. I.R. Cohen. Immune system computation and the immunological homunculus. *MoDELS 2006*, in O. Niestrasz et al. (Eds). Springer-Verlag, Berlin, pp. 499-512, 2006.
8. I.R. Cohen. Real and artificial immune systems: computing the state of the body. *Nat Rev Immunol.* 2007 Jul;**7**(7):569-74.
9. I.R. Cohen and D. Harel. Explaining a Complex Living System: Dynamics, Multiscaling and Emergence. *J. Royal Society Interface* 2007; **4**:175-182.
10. I.R. Cohen., U. Hershberg and S. Solomon. Antigen-receptor degeneracy and immunological paradigms. *Mol Immunol.* 2004 Feb;**40**(14-15):993-6.
11. I.R. Cohen. and D.B. Young. Autoimmunity, microbial immunity and the immunological homunculus. *Immunol Today* 1991; **12**(4):105-10.
12. S. Efroni, D. Harel and I.R. Cohen. Towards Rigorous Comprehension of Biological Complexity: Modeling, Execution and Visualization of Thymic T Cell Maturation. *Genome Research* 2003; **13**:2485-2497.
13. S. Efroni, D. Harel and I.R. Cohen. Emergent Dynamics of Thymocyte Development and Lineage Determination. *PLOS Computational Biology* 2007; **3**(1):127-136.
14. Z. Grossman and I.R. Cohen. A theoretical analysis of the phenotypic expression of immune response genes. *Eur J Immunol* 1980; **10**(8):633-40.

15. D. Harel. Statecharts: A Visual Formalism for Complex Systems. *Sci. Comput. Programming* 1987; **8**:231-274. (Preliminary version: Technical Report CS84-05, The Weizmann Institute of Science, Rehovot, Israel, February 1984.)
16. D. Harel. On Visual Formalisms. *Comm. Assoc. Comput. Mach.* 1988; **31** (5):231-274.
17. D. Harel. A Grand Challenge for Computing: Full Reactive Modeling of a Multi-Cellular Animal. *Bulletin of the EATCS*, European Association for Theoretical Computer Science 2003; **81**:226-235. (Early version prepared for the *UK Workshop on Grand Challenges in Computing Research*, 2002. Reprinted in *Current Trends in Theoretical Computer Science: The Challenge of the New Century*, Vol I (Paun, Rozenberg and Salomaa, eds.), World Scientific, pp. 559-568, 2004.)
18. D. Harel. On Comprehensive and Realistic Modeling: Some ruminations on the what, the how and the why. *Clinical and Investigative Medicine* 2005; **28** (6):334-337.
19. D. Harel. A Turing-Like Test for Biological Modeling. *Nature Biotechnology* 2005; **25**:495-496.
20. D. Harel. Statecharts in the Making: A Personal Account. *Comm. Assoc. Comput. Mach.* 2009; **52** (3):67-75. (Full version in *Proc. 3rd ACM SIGPLAN History of Programming Languages Conference* (*HOPL III*), June 2007.)
21. D. Harel and A. Pnueli. On the Development of Reactive Systems. in *Logics and Models of Concurrent Systems.* (K. R. Apt, ed.), NATO ASI Series, Vol. F-13, Springer-Verlag, New York, 1985, pp. 477-498.
22. N. Kam, I.R. Cohen and D. Harel. The Immune System as a Reactive System: Modeling T Cell Activation with Statecharts. *Proc. Visual Languages and Formal Methods,* (*VLFM'01*), 2001, pp. 15-22 part of *IEEE Symp. on Human-Centric Computing* (*HCC'01*).
23. H. Kugler1, A. Larjo and D. Harel. Biocharts: A Visual Formalism for Complex Biological Systems. to appear.
24. Y. Louzoun, H. Atlan and I.R. Cohen. Modeling the Influence of TH1- and TH2-type Cells in Autoimmune Diseases. *J Autoimmun* 2001; **17**(4):311-21.
25. Y. Louzoun, S. Solomon, H. Atlan and I.R. Cohen. Modeling complexity in biology. *Physica A* 2001; **297**: 242-252.
26. Y. Louzoun, S. Solomon, H. Atlan and I.R. Cohen. Proliferation and competition in discrete biological systems. *Bull Math Biol.* 2003 May;**65**(3):375-96.
27. L.A. Segel, E. Jager, D. Elias and I.R. Cohen. A quantitative model of autoimmune disease and T-cell vaccination: does more mean less? *Immunol Today* 1995; **16**(2):80-4.
28. Y. Setty, I. R. Cohen, Y, Dor and D. Harel. Four-Dimensional Realistic Modeling of Pancreatic Organogenesis. *Proc. Natl. Acad. Sci.* 2008; **105**(51):20374-20379.
29. N. Swerdlin, I.R. Cohen and D. Harel. The Lymph Node B Cell Immune Response: Dynamic Analysis in-silico. *Proceedings of the IEEE* (special issue on Computational System Biology) 2008; **96**(8):1421-1443.