

# LEVERAGING ALGEBRA AND LOGIC TO MODEL BIOLOGICAL SYSTEMS

RUCHIRA S. DATTA

## 1. INTRODUCTION

Mathematics is often said to be “unreasonably effective” for modeling reality. This phrase mirrors the feeling of surprise when a particular area of mathematics, glittering with crystalline beauty, suddenly turns out to be useful. Such surprises have sprung forth again and again, yet the sense of wonder endures. “Nothing is too beautiful to be true.”

Yet during the twentieth century, a curious phenomenon arose in the study of mathematics, particularly in the West. Whenever a branch of mathematics became overly useful, it was expelled from the main body of mathematics. The word *expelled* is not a rhetorical flourish; it was not simply a matter of a few mathematicians taking a “purer-than-thou” attitude towards some of their colleagues. The real manifestation of this expulsion was that practitioners of the new branch would migrate to a newly formed academic department with a new name, and new journals would be created to publish the work of this new community. Worst of all, the new branch would no longer be part of the postsecondary mathematics curriculum (if it ever was), and so in the usual course of study a budding mathematician would not learn about it except through serendipity.

The first branch to be so cast out was statistics; then operations research; then computer science; and now bioinformatics. When I worked as a software developer for a thermal analysis software vendor, my supervisor (one of its founders who had a doctorate in physics) was surprised to discover that, despite attaining doctoral candidacy in mathematics, I had never taken a statistics course since high school. I was relieved when another founder jumped to my defense: “Hey! That’s not math!”

*Applied mathematics* has a curious place in this story. In universities, it is sometimes a separate department from mathematics, sometimes not. But in any case, among the technocracy—those who create mathematics and those who use it—the implicit consensus is that applied mathematics consists of numerical analysis and differential equations. Although beautiful textbooks have been written elucidating each of these theories with nary an application in sight, these branches are recognized to be applied by their very nature, and their practitioners—whether in the applied mathematics department or not—are *expected* to be concerned with applications at least some of the time. Perhaps it is because these branches are born from that field which is most unquestionably “real mathematics”, functional analysis, that uniquely in this instance the field of their application has been allowed to retain the word “mathematics” in its name and, in some cases, to remain in the same department. For these mathematicians, theory and application form a continuous spectrum rather than residing in discrete compartments.

The obverse face of this coin, however, is that applied mathematics is often unconsciously assumed to consist *only* of numerical analysis and differential equations. Scientists and engineers seeking mathematical solutions to their problems tend to seek them first from these areas. This tendency has a strong historical foundation. Tycho Brahe supplied Kepler with much high-quality data about the motion of the planets. It was very fortunate that (unbeknownst to Kepler) an inverse square law governed this system, and the planets moved like point masses through the vacuum under the influence of a single massive body (rather than, say, in a binary star system), so that Kepler was able to make the beautiful discovery that the planetary orbits were elliptical. Newton's explanation of this fact was the triumph which ushered in the reigning era of differential equations. Again, it was very fortunate that the differential equation which introduced calculus to the world was one of the few that can be solved in closed form. In succeeding centuries, the wealth of phenomena explained by such masters as Lagrange, Laplace, and the Bernoullis made it seem that all would soon yield to the power of these methods.

Fourier showed the way to analyze *linear* PDE systems by exploiting the power of superposition. Of course, students of nature soon ran into differential equations systems which could not be solved in closed form. But such a system can often be analyzed as a system that *can* be solved in closed form, perturbed by a small correction—say, an oscillatory term. When these methods fail, numerical analysis steps in. Discretizing the state space into a grid or mesh and numerically solving systems of partial differential equations has now become the most widely used methodology in applied mathematics. It is mature, well-understood, and computationally efficient. These reasons have caused it to dominate.

This means, however, that this particular vein has been well-mined. By contrast, relatively few workers have attempted to shine the light of algebra and combinatorics on any given area of application. These powerful and beautiful techniques have great potential to yield truly new solutions to the problems of today. Here we very briefly survey a few papers which illustrate the promise of applying algebra and logic to the modeling of biological systems. Our aim is to delineate an area which might be called “symbolic biology” and determine its current state.

## 2. LINEAR MODELING

We begin with the paper “Linear Modeling of Genetic Networks from Experimental Data” by Someren, Wessels, and Reinders [SWR00]. These authors point out that large numbers of variables arise in modeling biological systems, and relatively few measurements of all these variables are available. To be more precise, the number of measurements is insufficient to estimate the parameters of a statistical model. In this situation, they point out that the sensible course is to use models with very few parameters. When we are in the dark about how to model the phenomenon, it makes sense to use a model which is as flexible and simple as possible. Once we have a model which fits, we may be able to decompose it into various domains. With insight into what is happening in each specific domain, we can use a specialized model that is tailored for that domain. We would use a particular differential equation system once we see that it is physically appropriate rather than assuming such a system applies and overfitting it to an insufficient dataset.

The authors hierarchically cluster the signals, in this case the pattern of gene expression levels from microarray data over various input conditions, into groups and

produce one prototype signal in each group. This serves to reduce the number of variables to be considered. Then they model the system as a linear network. The value of a variable at a particular time-point is a linear function of the values of all the variables at the previous time-point.

It should be noted that they threshold and normalize all the variables. Thus an affine transformation has been applied. So equivalently, the value of a variable at a particular time-point is an affine function of the values of all the variables at the previous time-point.

Furthermore, although they refer to a “network”, they do not seem to have actually used a nontrivial network. They do not explicitly define the graph in question, and indeed in their case it is the complete graph. The directed graph arises naturally from

$$x_j(t) = r_{0,j} + \sum_{i=1}^N r_{i,j} \cdot x_i(t-1),$$

the dynamic equation. It has  $N$  nodes, with an edge from node  $i$  to node  $j$  whenever  $r_{i,j} \neq 0$ .

Perhaps the reason they did not define this graph is that in the case they tried, a particular yeast gene expression dataset, none of the parameters  $r_{i,j}$  was exactly zero. Of course, this would be too much to expect from a noisy dataset. However, Figure 9 of the paper, which depicts the matrix of parameters they found graphically in terms of signs and magnitudes of the entries, is quite intriguing. Many of the entries in this matrix are very small. The logical next step would be to approximate these by zero. In this way we could gain insight into the structure of the regulatory network, and ideally we could decompose it into functional modules.

### 3. POLYNOMIAL MODELING

The above linear network is a special case of a polynomial network, one in which the value of a variable at each time point is a polynomial function of the values of its parent variables at each previous time point. Such networks are studied by Laubenbacher and Stigler in their paper, “Polynomial Models for Biochemical Networks” [LS03].

A method that is generally used to reduce the complexity of a network model is to discretize the set of possible values of the variables. The number of discretization levels is at the discretion of the modeler. In fact almost every model in use these days is discretized, whether explicitly or implicitly. In the case of a differential equations model the number of discretization levels is relatively large, perhaps up to the number of representable machine floats. Many modelers of biological systems these days have also looked at the opposite extreme, Boolean network models.

The number of discretization levels forms a continuous spectrum, from two states to the continuum. Laubenbacher and Stigler point out that if we choose the numbers of discretization levels to be primes, then the state space for each variable has the structure of a finite field. *Any* map of finite fields is given by polynomials.

Now the entire apparatus of polynomial algebra can be brought into play. This has been a very active field of research over the past two decades. In particular, the set of all models satisfying some constraints has a particular structure which can be studied with the tools of algebraic geometry. This set forms a geometric object called an *algebraic variety*, and computational tools exist to decompose this object into its components of various dimensions, called *irreducible algebraic varieties*.

Biologically meaningful constraints, such as “we know that this molecule regulates that molecule somehow”, or “the concentration of this compound is too low, and the spatial separation is too great, for it to affect that compound directly,” can be translated into statements about the network. Polynomial algebra can then be used to see the effects on the space of all possible models.

Of course, if the conclusions so drawn depended on the particular prime chosen then they would be physically meaningless. An important theorem is that as the number of discretization levels grows large enough, the polynomial networks arising from different choices of primes become equivalent.

Reinhard Laubenbacher has personally communicated that they have indeed modeled specific biological systems using polynomial networks. But we have not yet been able to obtain the report of their results.

A particularly intriguing observation these authors make is that in the special case when the polynomials are linear, the behavior of the dynamical system can be analyzed in closed form. They cite a preprint by Rene A. Hernandez Toledo of the University of Puerto Rico at Cayey [Tol], which we have not yet been able to obtain. Applying such an analysis in the framework of Someren et. al. might be very fruitful.

#### 4. HYBRID AUTOMATA

We now turn to the paper “Taming the Complexity of Biochemical Models through Bisimulation and Collapsing: Theory and Practice”, by Antoniotti, Mishra, Piazza, Policriti, and Simeoni [AMPSS03]. These authors use hybrid automata to model biological networks. This approach is taken by a few other groups, namely Rajeev Alur’s group at the University of Pennsylvania and Claire Tomlin’s group at Stanford.

In the usual automaton model, the state of a system is represented by a node in a graph, and transitions from state to state occur either deterministically or stochastically, possibly based on input controls. In a hybrid automaton, the state of the system at any given time is represented by both discrete variables (which may be thought of as specifying a node of a graph as above) and continuous variables. The continuous variables evolve according to some dynamical system. When the value of a continuous variable is about to violate a certain constraint, called a *guard* (for example, it reaches a threshold level), then the discrete variables jump to new values (so a transition takes place in the graph). This may occur either deterministically or stochastically. Once in the new state, the continuous variables begin evolving again according to a (possibly different) dynamical system.

We would argue that, in the contemporary context and particularly in modeling biological systems, numerical analysis and differential equations should no longer constitute the default paradigm for mathematical modeling. Rather, hybrid automata form a framework which subsumes and generalizes the former paradigm, integrates many other approaches, and includes powerful analytical tools. Within the last decade the engineering community has made this a very active field of research, but it is fair to say that hybrid automata are still not widely known among scientists and mathematicians.

Furthermore, even those who study them often think of hybrid automaton models as numerical PDE models with a few modifications because the numerical PDE models don’t quite fit. *Linear hybrid automata* have many nice properties which make their analysis particularly tractable, but because piecewise linear models are considered

stepsisters to “real” (numerical PDE) models, useful mainly for back-of-the-envelope analysis to gain insight, linear hybrid automata might be seen as a theoretical curiosity. When we instead take hybrid automata to be the fundamental tool in our toolbox (always keeping in mind, of course, that any model is only an approximation to reality), we conclude that we should generally first construct a model using linear hybrid automata and push it to its limits, only introducing higher-order approximations (such as numerical PDE) into our hybrid automaton model where justified and necessary for the problem at hand.

In the paper in question, the authors obtain the hybrid automaton by simulating the system with some other model, and then fitting piecewise linear polynomials to the traces so obtained. They do this so they can use the apparatus of *model checking* to answer queries about the system expressed in the language of *temporal logic*. The tool HYTECH, developed by Henzinger and others at Berkeley, can be used to answer queries about linear hybrid automata exactly. Such queries may be intuitively meaningful, for example, “is it *always* true that after applying this impulsive stimulus, this particular variable eventually returns to its equilibrium value?”

These authors use the hybrid automaton model primarily to enable such database queries, rather than as their primary model of the system. They next would like to create a specialized wavelet basis for the same purpose. They express a sentiment we have seen again and again: of course differential equations models would be more realistic since the variables of the system are continuous, but they are using piecewise linear models to approximate reality. (The other groups mentioned above are in fact using differential hybrid automata, that is, where the continuous dynamical systems are modeled by differential equations.)

For a dynamical system which is almost unknown, however, we first need to learn qualitative, which is to say topological, information, and the topology can be equally well-represented by a piecewise linear model as a differential equations one. In mathematical textbooks, piecewise linear models are often used to gain insight into a phenomenon in a case-based analysis, but such reasoning is quickly dropped. Models which would actually fit the data well cannot occur because the many cases needed would become too cumbersome. In reality, however, we have computers to deal with cases. An important fact to keep in mind is that when we solve differential equations systems numerically, we generally discretize the state space into a polyhedral mesh, and represent the state function within each cell by its Taylor polynomial of a fixed order. Thus, in fact, we simulate a hybrid automaton. So differential equations models end up as very large hybrid automaton models which are piecewise linear (or quadratic, or polynomial of an order equal to the number of derivatives we are keeping track of). Without a particular reason to adopt a differential equations model (for example, knowledge of the physical nature of the system), it makes sense to drop the restriction that the hybrid automaton model must be of this special kind and instead look for one which is as small as possible while still fitting the data.

Furthermore, hybrid automata are also flexible enough to accommodate stochastic processes. These may be more appropriate to model some biological phenomena where the concentrations are very small. Decomposition into spatial compartments and functional modules can be modeled by *interacting* hybrid automata, which may be *composed* of smaller ones. *Refinement* of our knowledge about the system can be modeled by refinement of automata. All these terms have precise meanings in

systems theory, and the engineering community has developed extensive tools for their analysis.

We visited NYU and saw a demonstration of the tool SIMPATHICA that Mishra's team has developed, integrating existing tools to simulate models and using HYTECH to answer queries. An intriguing observation was made by Nadia Ugel, a student of Mishra who is one of the authors of SIMPATHICA. The dynamical equations of chemical kinetics involve the concentration of some catalyst raised to some power. This exponent is usually one, sometimes two, rarely three, and almost never four. (Their group, on the other hand, is modeling this exponent as a real number.) We may be able to exploit this observation when modeling such systems algebraically.

## 5. SYSTEM IDENTIFICATION USING ALGEBRAIC GEOMETRY

Finally we turn to the papers "Generalized Principal Components Analysis", by Vidal, Ma, and Sastry [VMS03], and "An Algebraic Geometric Approach to the Identification of Linear Hybrid Systems", by Vidal, Soatto and Sastry [VSS03]. As explained above regarding the work of Someren et. al., identifying a system naturally falls into two stages: clustering and estimation. These authors use algebraic geometry to perform both these functions simultaneously. They have been applying their technique with good results to problems of machine vision.

The basic idea is most easily explained for piecewise constant models. In this case the number of clusters is the degree of a polynomial, and the constants (the values taken on by the state variables) are the roots of this polynomial. Thus system identification becomes polynomial factorization. To find the degree of the polynomial, they form a Vandermonde matrix from the data. The number of clusters is the smallest number such that the Vandermonde matrix of this dimension has full rank.

For piecewise linear data, the problem becomes one of identifying linear subspaces of Euclidean space into which the data falls. They embed the data into a space of higher dimension using the Veronese map, which maps a vector of variables to the vector of homogeneous monomials of a higher degree. (They seem to have chosen this map because it was easy to understand, but other maps might be used with nicer properties.) In the higher dimensional space they randomly project the data onto a single linear subspace. There they have a product of linear forms, rather than a univariate polynomial as they had before.

Of course, real data are noisy. So instead of expecting these algebraic relations to hold exactly, they use Lagrange optimization to minimize the deviation of the data from the product of linear forms in question.

This is an exciting area with much mathematical work yet to be done. Given the promise of linear hybrid systems, we are eager to apply this technique to the identification of systems arising in biology.

## 6. CONCLUSION

In this project we have learned about several intriguing directions for modeling biological systems. The teams who are applying these various tools of algebra and logic are very few, and none of them has synthesized all of these approaches. As far as we could determine, only Laubenbacher's group is using the powerful techniques of algebra. (We would like to acknowledge funding by Laubenbacher, NSF grant DMS 0138323, during this work.) We would next like to use these techniques in concert to model an actual biological system.

- [AMPSS03] M. ANTONIOTTI, B. MISHRA, C. PIAZZA, A. POLICRITI, and M. SIMEONI, Taming the Complexity of Biochemical Models through Bisimulation and Collapsing: Theory and Practice, *Theoretical Computer Science*, 2003.  
<http://www.cs.nyu.edu/cs/faculty/mishra/PUBLICATIONS/03.tamebisim.pdf>
- [LS03] R. LAUBENBACHER and B. STIGLER, Polynomial Models for Biochemical Networks, preprint.
- [SWR00] E. P. VAN SOMEREN, L. F. A. WESSELS, and M. J. T. REINDERS, Linear Modeling of Genetic Networks from Experimental Data, *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB 2000)*, La Jolla, California, 2000.  
<http://www-it.et.tudelft.nl/itbibliography/reports/2000/conf/Somer00b.ps>
- [Tol] RENE A. HERNANDEZ TOLEDO, Linear finite dynamical systems, preprint.
- [VMS03] R. VIDAL, Y. MA and S. SASTRY, Generalized Principal Component Analysis, IEEE Conference on Computer Vision and Pattern Recognition, 2003.
- [VSS03] R. VIDAL, S. SOATTO and S. SASTRY, An Algebraic Geometric Approach to the Identification of Linear Hybrid Systems, IEEE Conference on Decision and Control, 2003.