At the Interface of Biology and Computation

Alex S. Taylor[1], Nir Piterman[2], Samin Ishtiaq[1], Jasmin Fisher[1]
Byron Cook[1], Caitlin Cockerton[1], Sam Bourton[3], David Benque[4]

[1] Microsoft Research
21 Station Road,
Cambridge, CB1 2FB, UK
[ast; sishtiaq; jfisher; bycook,
v-caitc]@microsoft.com

[2] Department of Computer Science
University of Leicester
University Road
Leicester, LE1 7RH, UK
nir.pieterman@le.ac.uk

ABSTRACT

Representing a new class of tool for biological modeling, *Bio Model Analyzer* (BMA) uses sophisticated computational techniques to determine *stabilization* in cellular networks. This paper presents designs aimed at easing the problems that can arise when such techniques—using distinct approaches to conceptualizing networks—are applied in biology. The work also engages with more fundamental issues being discussed in the philosophy of science and science studies. It shows how scientific ways of knowing are constituted in routine interactions with tools like BMA, where the emphasis is on the practical business at hand, even when seemingly deep conceptual problems exist. For design, this perspective refigures the frictions raised when computation is used to model biology. Rather than obstacles, they can be seen as opportunities for opening up different ways of knowing.

Author Keywords

Computational biology, ethnography, philosophy of science, science studies, epistemology, materiality.

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms

Design; Human Factors.

INTRODUCTION

Recent advances in computer science (CS) have led to a new class of tool for reasoning about the theoretical behaviors of complex and infinite systems. Unfortunately, these powerful computational tools can require some fundamentally different ways of conceptualizing and approaching established problems. The kind and format of questions that can be asked present several difficulties for end users, as do the results the tools produce. In this paper, we describe an ongoing project in which we are designing the user interface for one such tool and our efforts to tackle some of the

attendant difficulties. The tool, called *Bio Model Analyzer* (*BMA*), supports the modeling and analysis of what in biology are called gene regulatory networks.

Surprisingly, the new wave of computational tools being pioneered in CS and operationalized in the sciences has been a neglected topic in HCI. Research has focused on the collaborative aspects of scientific (and in some cases biological) tools, often looking to support the sharing of data, analysis and results (e.g. [4,5]). However, the role computation plays in interaction and design has had little if any immediate attention. This absence is understandable given the depth of technical understanding demanded and, it must be said, the apparent dryness of the subject area. However, as we'll show, the peculiarities of computational techniques do have consequences for interaction, and these seem particularly relevant to design and HCI in light of the claims coming from some quarters—somewhat grandiose, perhaps—of biology presenting the next wave of technology [7].

The design of BMA, a software tool that determines the stability of biological networks, is thus offered as a way into thinking about the emerging role computation is playing in biology and the sciences more broadly, and how HCI might usefully contribute. Our focus details attempts to design a tool that appeals to a range of users, from the computationally savvy to experimental biologists with no background in computer programming. It also, however, raises what we see to be some fundamental issues with wider implications. It animates a deeper discussion around some of the tensions or frictions that can arise when new computational techniques are introduced into established scientific practices, and what these frictions might mean for tool design.

Succinctly, the contributions we wish to draw out of this work can be summarized in three points. The first is narrow addressing, specifically, the design of interactive tools for computational biology. It aims to show there are good reasons for displaying the multiple perspectives that appear intrinsic to the new class of computational tools—tools like BMA—even though they can introduce conceptual tensions. The other two contributions open the work up to the broader relations between HCI and the sciences. With the second, we'll aim to show that HCI has an important con-

3 Quantum Black, 4-5 Bonhill, London EC2A 4BX, UK
4 Royal College of Art, Kensington Gore, London, SW7 2EU, UK

tribution to make in the uptake of computation in the sciences, and crucially how computation becomes entangled in the ways scientific knowledge is constituted. We'll show that HCI has the potential to intervene at the intersections of computation and science, where new findings are being made, but also where the frictions between users and tools can be at their most acute. Relatedly, the third contribution is to tie HCI's potential interventions into broader questions about how tools prioritize certain ways of knowing. BMA aims to make visible the conceptual frictions that can arise in computational biology by allowing results to be seen from heterogeneous perspectives. Thus it aims for an openness to ways of knowing. For HCI, we hope to invite questions about design's role in achieving such an objective.

To begin, we first outline the empirical materials that have contributed to this paper and also give some background to the theoretical arguments that have informed it. We then discuss computation in biology and describe BMA's computational underpinnings. The body of the paper turns to a problem that has been central to BMA's design, one we've called the *proof-mode problem*. At this stage, suffice to say this is something that has been repeatedly returned to during the tool's development and provoked much discussion amongst the project members. Having discussed this problem from two particular viewpoints, we will then return to the paper's overall contributions in an implications and general conclusions section.

Research Approach and Orientation

The presented work is taken from an extended and on-going project in which a core team of eight members from biology (1), CS (2), social science (2), design (1) and software engineering (2) have collaborated on the implementation and design of BMA. Specifically, we use materials from a design phase spanning eighteen months, in which: group members regularly attended (bi-weekly and sometimes weekly) meetings to discuss the tool's design; preliminary one-on-one interviews were undertaken with six biologists invited to test BMA and give detailed opinions about its design and purpose; and two non-team members trialed the tool, intensively, in their own research. Alongside this, a number of iterations of the tool were internally implemented and trialed, with numerous features added and modified.

For the purposes of this paper, dominant use is made of the in-house, one-on-one interviews with the team's members, the transcripts and field notes taken in team meetings, and interviews and written materials from one of the non-team members who participated in a full trial of BMA. Although the general thrust of the comments from the early, one-on-one interviews with the six biologists is captured, only passing reference is made to the extensive transcripts we compiled. Together, the materials used are assembled to construct what we call two 'viewpoints'. Both of these elaborate on particular aspects of the proof-mode problem, weaving together a story about the issues it raised and a design perspective it precipitated.

As this use of two viewpoints to study tool use might be seen to deviate from the norm in HCI, some explanation is called for. The approach has been broadly motivated by three factors. Most superficially, the first is to do with the limitations of space and thus an effort to choose the most informative way of introducing the tool and the kinds of problems faced. Secondly (and still practically motivated), the materials used have been chosen to string together a coherent narrative from what—even for the team members—has become a convoluted project. The focus on a singular problem will limit, it is hoped, the amount of technical terminology and domain specific knowledge needed to engage with the text. We're acutely aware that the technical details discussed below will be unfamiliar to many and thus our hope is the viewpoints will make the trajectory of our work and problems faced easier to follow. Thirdly and most significantly, the viewpoints and narrative have helped piece together a distinctive position on tool use and design, one we will turn to next. It should be said at this stage, however, that the position emerged from the interplay of empirical work and reading around the relevant topics in biology, the philosophy of science and science studies. The objective with this was not, purely, to prescribe functions and features for BMA, but to enliven a dialogue around the broader role of computational tools and an orientation to their design.

The thrust of the paper's arguments, then, draws on a literature concerned with how scientific knowledge is constituted. Much of this research has its origins in seminal work in science and technology studies such as Bowker and Star's investigations into the social character of scientific categories [6] and Pickering's insights into the complex and intertwined trajectories of knowledge production [24]. More particularly, though, our work has come to be shaped by recent research in feminist technoscience and a turn to a 'new' or socio- materialism [27]. Specific references will be made below, but the gist of this broad movement is that the separations between knowledge, knower and object/matter are part of a long-established but problematic metaphysics of scientific ways of knowing [2,28,31]. Accordingly, to really get to grips with how knowledge is constituted, it is argued one must see these three things as inexorably enmeshed, continually being assembled and forming particular relations. Knowledge is thus seen as a relational achievement, always in a state of becoming, of moving on.

Another strong influence has been from a related literature in science studies and the philosophy of science looking at the relations between practice and theory [15]. Helping to make sense of BMA's use has been a perspective that sees knowledge as part of a practical endeavor, where materially bound, pragmatic activities and decisions—not overarching abstract theories—are determinants of how scientists see, interact with and go on to make sense of their phenomena [15,18,25]. Furthermore, theory is not seen as the ultimate achievement, outside of or superior to science on the ground. The different materials and ideas scientists such as

biologists work with are viewed as continuously transforming in practice, being one moment the objects of study and another the tools or theories used to manipulate and understand them [18]. Again, the always-shifting relations are seen to be the source of these unfolding transformations.

Finally, the paper's arguments are set alongside a more general set of theory-oriented perspectives emerging in HCI in which interactive systems and the material interventions enacted by them—in practice—are being recognized as integral to ways of knowing and being [22,26,29,30,32]. Although the work we present is specific to science and biology, we see our arguments to be situated in this broader intellectual dialogue. Tentatively at this stage, they are intended to offer a way of framing interaction that moves the analytical concerns on from prefigured or essentialist notions of users, tools and the stable interfaces between them; a starting point is articulated that opens up how an assemblage of multiple agencies and emergent entanglements can be investigated, (re)configured and (re)designed.

BMA AND COMPUTATIONAL BIOLOGY

These discussions and how they have provided routes into thinking differently about BMA's design will be returned to in a moment. First, let us backtrack a little and provide some context.

Computation is playing an increasingly significant role in modeling and analyzing biological systems. Systems biology, for instance, attracts those from diverse disciplines such as biology, chemistry, physics, mathematics, electrical engineering and computer science, among others. Harnessing computation, it aims to explain how organisms function by modeling and analyzing the behaviors and relationships of genes, proteins and other biological elements [12]. Tools such as *CellDesigner™* [13] and *Virtual Cell* allow cell network processes to be graphically drawn and simulated, placing an emphasis on the modeling of stepwise processes along biological pathways (i.e., the sequence of molecular events that lead to cellular behavior).

The point has been made well elsewhere that such *in silico* techniques offer ways of abstracting the data and making inferences that obviate the need to manually work through every possible case of a problem (and, for that matter, resort to time consuming and costly experimentation in the lab). Predictably, though, as the area has matured and the requirements have become more specific, distinct computational techniques have emerged to tackle narrowly defined problems. For the purposes of this paper, a helpful way to think about the spectrum of computational techniques is to separate them into *bottom-up functional* and *top-down structural* approaches [12,19].

Bottom-up approaches—seen in software like *CellDesigner* and *Virtual Cell*—construct models directly from experimental evidence, building up representations from discrete biological functions (hence bottom-up functional). Ostensibly, the model represents the way a biological organism

works or functions—it *simulates* the biology. So model parts may be cells, genes and/or proteins and the model may, for instance, represent the signaling pathways between them. By necessity, there are omissions, simplifications, and abstractions made in the models, but the aim is to instantiate a one-to-one mapping with the elements of a biological network and the relationships between them.

Top-down techniques employ a significantly different meaning of model. Models may or may not have an immediate correspondence to biological processes. A high-level structure is determined by the computational technique applied, in a sense how the code or program is run to produce them. A model may thus execute a proof, where the relations between a biological elements are systematically checked against testable criteria [11]. The structure is thus imposed top-down by the way particular relationships between entities are formalized and evaluated, programmatically. The key point to recognize here is that the model, and in particular its structure, is derived top-down by the execution of the proof. Again, this contrasts with the bottom-up functional approaches that incrementally build up models based on the way the biology functions.

BMA is an example of this latter approach to modeling. The computational underpinnings of the tool runs what is known in CS as a symbolic model-checking algorithm to prove whether the model achieves *stabilization* [8,9]. It checks, in other words, whether the modeled system will, from every possible starting point, always eventually reach a single 'safe' configuration or equilibrium. In biology, such modeling could offer, for example, a means of assessing whether a cellular system develops in a stable way that is robust to perturbations from the outside environment, and experimenting with interventions that might improve such a robustness.

The innovation here is that BMA is able to prove stabilization for highly complex models that may have a huge number of configurations or states. To work around problems of scale, the tool evaluates symbolic sets of this extremely large state space. Furthermore, rather than analyzing the whole network, it answers questions about discrete components such that stabilization is established as a composition of smaller mathematical arguments (i.e., lemmas). In effect, BMA's analysis results in the quick discovery of components that take initial steps towards stabilization and then determines the mechanism by which other components follow that lead.

It is this that defines the tool's approach as top-down. The structural configurations and sequential path through them are determined not by biological functions *per se*, but by the way the underlying algorithm works. As a matter of fact, the algorithm has its origins in techniques used for model-checking computer code and testing whether, given a particular arrangement of properties, a software program "terminates" without errors or 'bugs' [see 10]. Consequently, the algorithm and its incremental progresses through the
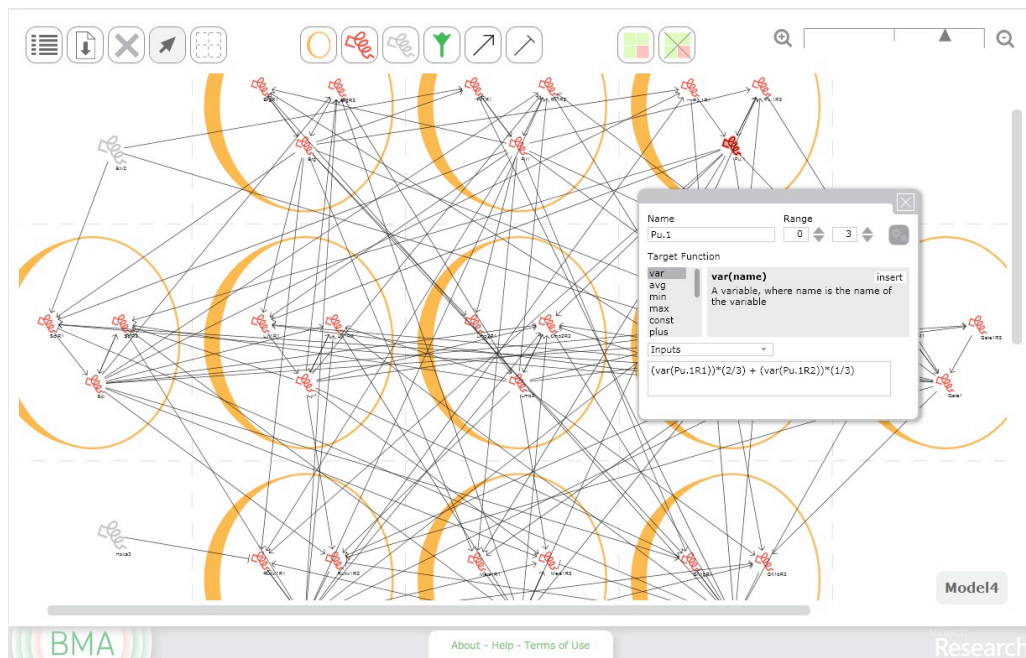
Figure 1. BMA 'mode' for constructing and manipulating regulatory network, with variable panel open for editing parameters.

model has what could be best described as a loose correspondence to the biology and treats it more like a generic network of nodes and connections.

The Proof-mode problem

The computational technique that underpins the BMA tool has been more or less complete since 2011 and shown to successfully replicate the results observed in laboratory experiments with mammalian skin cells and *Caenorhabditis elegans* (*c. elegans*) development [9]. The current emphasis in the project is to develop a graphical user interface so that biologists with little to no programming expertise can use the tool. To this end, a web-based front-end has been designed to support, one, the graphical construction and manipulation of biological networks (Fig. 1) and, two, the execution of the proof and graphical presentation of the results (Fig. 2). (For details of the UI see [3]). Although some sticking points persist, some of which we will come to later, the first of these UI modes has reached a relatively mature stage of implementation. In both our preliminary interviews and trials, respondents found the features for constructing and manipulating the biological networks straightforward to use and compared them, favorably, to familiar graphical layout applications such as Microsoft PowerPoint and Adobe Photoshop.

Less developed is the way the tool displays the results. The most recent complete implementation runs the analysis with the straightforward selection of a 'proof' button in the tool's interface. However, the display of results merely indicates whether the network has stabilized (green highlight) or, if there is instability, which network elements have failed to stabilize (red highlight)(Fig. 2). To add to the paucity of this visual representation, the red highlight may also

indicate the analysis having "timed out". This is because the output from the proof originally treated the timing out of the analysis as equivalent to instability. Thus the tool effectively produces a binary output for each element in the model, representing stability or instability. Biologists responding to this part of the tool have been confused by these results and unclear how they might be applied.
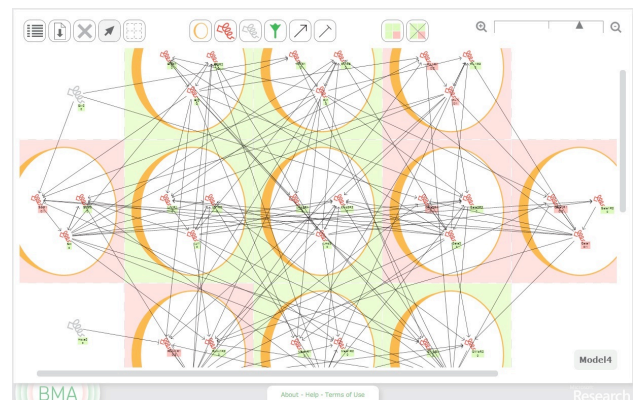
Figure 2. Results of analysis in BMA's 'proof mode' with stability of variables highlighted in green and instability in red.

This evident weakness in the tool, that has come to be known by the project's members as the 'proof-mode problem', has been revisited throughout the work. In project meetings, a host of different designs for representing the proof and its output have been proposed and rejected. On the face of it, the difficulty appears to be the result of the top-down structural approach used in the tool. For the project team, the protracted discussions have, time and again, resorted to, on the one hand, the lead biologist struggling to understand why some aspect of the biological processes

can't be represented and, on the other hand, the computer scientists explaining that the computational approach employed doesn't allow for the simulation of biological processes. All too often the outcome of these discussions has been to move on to other less problematic features of the tool's design. It isn't that either side is unaware of the constraints. On the contrary, there is a high degree of understanding of both the biology and computational workings from both sides. At issue appears to be the seemingly intractable problem of visualizing the tool's analysis in a way that usefully represents the biology.

It is worth noting, too, that this tension surfaces with most vigor in the design discussions of the tool's interface. Before the UI design was considered, it seemed the primary achievement was simply to reproduce the results of empirical experimentation. And yet the interface design introduces problems because it situates the tool in use. It demands that beyond demonstrating BMA is capable of replicating the results of experiments, there must be some recognition of the sense-making activities of biologists. Thus the work at and on BMA's interface appears to be about more than designing an easy to use tool. It is through such a concern that we see the multi-disciplined project team coming to terms with what they are developing and how it is of practical use in scientific research. The work at and on the human-machine interface, then, cannot be reduced to, as Suchman richly words it "machinic operations" but on the contrary opens up to inspection "a specifically enacted site of extended, heterogeneously constituted human/nonhuman capacities for thought and action." [27]. Exposed, that is, are the entangled relations between user and tool and these, in turn, invite questions about the doings and knowings enacted through them.

In the following we tease this out a little more, with the aim of illustrating how it is that scientific practices can be further understood and designed for. We recount an issue that revealed itself in our efforts to deal with the computational approach used in BMA and specifically one that arose in tackling the proof-mode problem. Our inclination, early on, was to think of the problem as a weakness in representation: that the tool, because of its top-down approach, misrepresented the biology. We found ourselves adopting what could be fairly described as a realist position, with the implication that there are absolute phenomena to be measured in biology, and, if you will, 'true' ways in which to represent them. This position has been given credence in the philosophy of science with discussions reinforcing the singular 'meaning' of biological phenomena and criticisms of computational techniques for being one-step removed from such 'truths' [14,19].

However, by paying close attention to the tool's use in our early trials, and particularly the efforts to overcome the proof-mode problem, we found the tool being used in practical ways to get on with the business at hand. In other words, the use of the tool and efforts to overcome the proof-

mode problem overrode the concerns we had for what was real or true in biology, and shifted the emphasis to the practical business of moving on. The biological models were not enlisted with any strong theoretical motivations or to determine any grand truths in science. Instead, as the historian and philosopher of science Evelyn Fox Keller has observed in remarkably similar work, the modeling seemed pragmatically useful for "designing new kinds of experiments, for posing new kinds of questions, and for guiding new kinds of manipulation of the system itself." [17:S82].

A VIEW OF USE

The first of our viewpoints onto the proof mode problem is centered on the first attempt by a non-member of the BMA team to construct a network and test a model using the tool. So far, two different case studies of this kind have been undertaken. The one considered here, lasting over 3 months and led by a MSc. student Lucy, presents the most discussed by the team and well-documented to date.

Lucy's master's studies are part of a wider program of work at an internationally renowned university's medical research institute. This program applies an "integrated approach" to determining the regulatory networks responsible for blood stem cell development. In biology, blood cell development or *haematopoiesis* constitutes one of the paradigm cases for studying stem cells' peculiar capacity to both self-renew and differentiate into other cell types. A considerable literature therefore exists on the subject and there is plenty of scope for mining extant results and cross-validation. Lucy's use of BMA is thus part of a continued effort to incorporate computational techniques into the institute's integrated approach—alongside other techniques such as *in vitro* experimentation and *high-throughput sequencing*. It also, though, serves as a test case for, (1), confirming BMA's capabilities for modeling known biological phenomena and, (2), determining whether the tool provides the basis for extending know-how in this discrete but important problem area of biology.

With respect to the first of these latter two objectives, Lucy's original aim was to determine whether a regulatory network of what are known as transcription factor proteins and their interconnections, modeled in BMA, exhibited blood cell behaviors observed in the laboratory. Although the number of different proteins involved in blood cell development is known to be in the thousands, a review of the literature had identified thirteen transcription factors as crucial to blood cell differentiation—as their absence precluded normal blood production—and these formed the basis for constructing the network model. Each protein was modeled qualitatively, varying between a small number of levels (e.g. 0-1=low to medium, or 1-2=medium to high). Also incorporated were the complex array of promoting and inhibiting connections. In combination, these interconnections and levels define, if you like, the transcription factors that can communicate with each other and the amount of influence they have over each other. Lucy's use of BMA

was to assess whether this network of proteins, connections and value ranges produced a model that was self-contained and did indeed stabilize as expected.

With respect to the second objective, Lucy's hope was that the introduction of perturbations into the BMA model might provide the basis for new hypotheses to be tested in the lab and ultimately new understandings of blood cell development. For instance, Leukemia is known to be associated with particular genes being 'knocked out'. The modeling in BMA was seen as a means of reproducing these knockouts and testing possible counterbalancing interventions, thereby hypothesizing about potential therapies.

Lucy's first problematic encounter with BMA was in configuring the transcription factors and connections in such a way that the proof could be executed. With her initial attempts to compose and test the model, she found that the tool timed out, repeatedly. With help from the team's computer scientists, this was discovered to be because the connections she'd defined between the proteins were too literal and consequently had led to a complexity that was too demanding for the tool. The details behind this are unnecessarily technical for the arguments here, though it is worth noting that to successfully execute the model, Lucy needed to change her approach to modeling by further abstracting the structural configuration to conform to a notion of a model expected by the tool. As Byron, one of the team's computer scientists put it, "she didn't [initially] use it in the spirit it was written" and this necessitated a change in approach with the "right" structure of modeling in mind. The biology, in effect, had to be re-configured, structurally, to be 'readable' in computational terms and this required a change in mindset—or "spirit"—on Lucy's part.

We begin, here, to get a sense of particular frictions that can arise when modeling biology using computational techniques and specifically when a top-down structural as opposed to bottom-up functional approach is applied. Such frictions further manifest themselves in other ostensible limitations to the tool. Although Lucy was eventually able to execute the blood cell network using BMA and prove it stabilized, she found a severe limitation of the tool was the way it accounted for (or to be more precise did not account for) the progression of the biology over time. Quite simply, the tool does not model time. Indeed, it intentionally avoids doing so. To handle highly complex cell systems, and work through an inordinately large number of possible states, BMA, by design, avoids having to simulate cellular development over time. In Byron's words:

> "So we're not running the system; we're more making an argument at a level that is higher than basic time... We just start thinking about how the system is configured and start making logical inferences that are valid about the behavior..."

Here again we are confronted by the peculiarities of the top-down approach applied to model checking in BMA. The "logical inferences" that are made are derived using algorithmic formalisms, not properties purely of the biology and

specifically not employing any notion of temporal progression of cellular organisms. In the case of Lucy's blood cell network, the proof did not model the five stages of cell differentiation that have been observed in experiments, nor did it assume any starting points or complete pathways intrinsic to the biology. In simple terms, it took into account the range of variation and tested locally (and algorithmically) defined sets of connections. In the following, Nir, another of the team's computer scientists, patiently explain this:

> "The way I think about [these] models is that they capture all points in time […] So in these kinds of models there is no element that says, well, I'm allowed to refer to my watch and change my behavior according to time […] I'm constructing here a static picture of how they change for every possible time and every possible value."

The point to emphasize here is that something ordinarily thought to be intrinsic to biology, time, is not accommodated in the modeling approach used by BMA [18]. The continuous and time-dependent properties of biological organisms (conventionally represented in biology using differential equations [18:250]) are quite simply absent. Conveyed by Nir is that the very power of the tool is achieved because it avoids having to work from any estimated starting points (that could be error-prone) and instead runs through and checks every possible state. So, in BMA, time is abstracted away, eliminated. Indeed, Lucy raises this as a source of confusion in explaining how she imagined using the tool:

> "We weren't really sure what was going to happen initially. We were going to try and get it to go from one start point to the next and then through the stages individually but we couldn't really do that once we'd abstracted everything so then we had to find another way of looking at the different points within it."

She goes onto explain:

> "We ended up using a program based on [the model] in Excel—that Nir created—because a lot of the time it's not actually the endpoint that's most interesting, it's how it got there and the intermediate stages—especially for this project—that we're interested in."

Lucy describes a workaround for the tool's incapacity to simulate the temporal changes within the blood cells. Having proved stability using the tool, she had Nir help put together an Excel table so that she could manually enter and test values in her paired-down model. This enabled her to see, from predefined starting values, how the model progressed. In effect, she required a means of re-introducing and simulating biological time and stepping through the five known stages of blood cell differentiation.

Once again, the use of BMA appears to exemplify the kinds of fundamental breakdowns that can occur between the representational forms considered intrinsic to biology on the one hand, and afforded by computation on the other. Lucy required a solution outside of the tool to re-introduce and observe cell-change over time, even if the temporal delineation was crude.

And yet the actual use of BMA in practice paints a very different picture. Unsurprisingly, perhaps, rather than in-

surmountable, such frictions on the ground seem to be treated as pragmatic ones (at least in the case of BMA). What appears to have been far more important to the tool's potential users—users such as Lucy as well as others we interviewed—has been its practical value. Lucy's comments on the immediate value of BMA lends some initial support to this:

> "[The tool] confirmed that there weren't any crucial links missing as well and that they weren't the wrong way round cause when we tried switching them it went completely wrong."

So the tool provides a quick if not completely foolproof way of assessing a model's functional connections. Furthermore, as Lucy explains, it sits within a more intertwined set of practices of judgment, understanding and iterative modifications to the model:

> "It is interesting cause you see which ones don't stabilize … you have to then go back and look at the biology and see if that is interesting […] [and if] we've missed some connections… Also because we are only using the 13 genes and there are hundreds actually involved in the process like sometimes it would be useful to have other inputs but that's a biological issue so we have to go back and say what is stopping that from stabilizing and it's often because that particular protein is also regulated by something else."

Such pragmatism evidently involves dealing with the limitations of what is known and working with the tools that enable one to "go back" and test out new or alternative possibilities. It seems, modeling tools such as BMA are unremarkably subsumed into an assembled "machinery of seeing" [1] or "reasoning machinery" [25:20], and, in doing so, are less impeded by problems that appear, at first glance, fundamental. As Lucy suggests, her motive is really to be able to revisit the model and/or the biology and test out different ideas to expand what is known. She is relatively unperturbed by, even disinterested in, whether the tools she uses are built on or result in conceptual inconsistencies.

So, from afar, BMA could well be seen as a source of conceptual difficulties (as they were by the team). Moreover, these difficulties or frictions are manifestly born out of the coming together of computation and biology and the representational forms they allow for. Yet, the difficulties, in practice, are centered on getting the results to make sense in practical terms, in short, in using them to determine "what to do", a way to move on. Moving between the various representations of the biology, they provided Lucy with a way to ask more questions about the model represented in BMA and to test-out changes in the blood cell network.

For BMA's design, the issue worth highlighting is how we should understand the output of the analysis. Seen alone as a static image indicating binary outcomes, it appears an object vulnerable to conceptual difficulties. However, situated in use—practical use—we see it, as far as possible, being implicated in and forming a basis for what science studies scholars Amann and Knorr Certina [1] have described as a *sequence of practice*; that is, contributing to "the work of seeing what the data consist of" [1:138] and

more particularly for BMA 'what to do with what is seen'. By interleaving the tool's model with a temporal representation in Excel, for instance, the user can see the analysis introducing a way to interrogate and further augment a continually evolving 'picture' of cellular development. The "[i]mage analyzing exchanges are not just "about" an object; they are also "with" an object." [1:140]. The fault lines seem not to hinge on the conceptual consistency of the data and their representations, a strict notion of what is real, or the coherence of the techniques used to produce and represent results. Rather it is whether they enable a trajectory of interpretation and intervention.

A VIEW FROM DESIGN
The second of the two viewpoints we want to consider, that from design, conveys similar points. However, it's also hoped they may introduce an approach for opening up the representations and enabling a heterogeneity of trajectories.

From the outset of the design work (as we've discussed), the apparent incongruity between the computational workings and the biological processes kept the team coming back to questions concerning how the analysis could be revealed to the user and, in particular, how a representation of cell development over time might be incorporated into the 'proof mode'. Much of the motivation for this came from feedback from the initial interviews with BMA users as well as from the two non-team members (including Lucy) who used the tool in anger. Also instrumental, was the team's lead biologist. She, over the course of the project, reiterated the importance of a temporal component in the tool and, if anything, became more vocal about its absence as the project progressed.
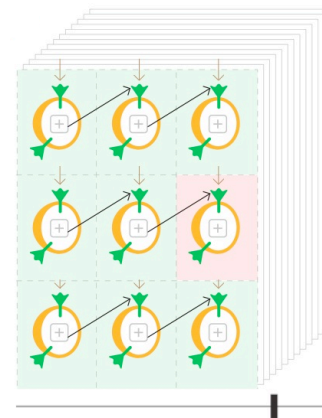
Figure 3. Sketch to visualize proof progression where the user can 'scrub' through pages or steps of the proof.

The proposal repeatedly returned to has been to visualize the temporal progression of the proof—that is, display to the user some representation of execution of the proof over time ('proof time', so to speak, as opposed to 'biological time'). Figure 3 shows a possibility for one such representation that has been envisioned and discussed by the team's members. However, ideas such as these have presented something of a dilemma for the team. The worry has been

that proof time might be confused with biological time; all members of the team have been sensitive to the different notions of time at play and weary about introducing any further confusion for users. Figure 3 illustrates how proof time could be a source of confusion; the problem is that the temporal ordering of the steps has no direct mapping to the actual progression of the biology. In strictly realist terms, it misrepresents what is known about the biology and worst still has the potential to lure users into thinking they are seeing something biologically meaningful. Thus, it's evident why representing proof time may have been a concern for the team; there appears an intrinsic conflict or tension.

Yet, the pragmatic perspective we've discussed above has provided a different way of approaching this problem and, as it happens, offered a more constructive starting point for design possibilities. In pragmatic terms, the problem becomes one of whether proof time can be visualized in such a way that allows users to see the analysis as part of a sequence of practice. That is, whether the representation of proof time enables one to take a next step—to move on—in one's trajectory of experimentation and idea generation. This, then, is to see design not as a means to insist on and prioritize a representation that conforms to a singular conception of biology, but, instead, as a means of opening up opportunities for further intervention.

At first glance, such concerns with BMA's interface may themselves seem to be purely pragmatic ones to do with the narrow problems of user interface design. Naturally, there are practical issues concerning visual layout, consistency, intelligibility of the UI, and so on. However, this work of implementing the UI brings us back, again, to questions of knowledge making in science and specifically the interdisciplinary frictions raised in bringing computation and biology together. Relevant to these issues are the ongoing discussions in science studies and the philosophy of science turning on the supposed separation between representation and intervention [15] or, to put it another way, knowing and doing. Such binaries have received considerable attention with scholarly research contesting and refiguring the seemingly incommensurable, such as the divides between subject-object, theory-practice, human-nonhuman, real-unreal, nature-society, etc. [2,15,20,21,28,31]

Although engaged in vastly different realms of scientific life, two relatively recent works provide particularly helpful places to start from here, one the book *Science and an African Logic* [31] by Helen Verran and the other by Karen Barad [2] in her detailed account of the unfolding controversies in quantum physics. Both present an analytic frame in which knower, knowledge and thing (matter) are inseparably entangled and located within ongoing, processual practices. The broad argument made is not whether scientific (or for that matter indigenous) ways of knowing have a special or privileged right to truth claims or reality, but rather how knowing is something enacted through entanglements of people and things in-action.

Verran, for example, weaves her way around and within the muddy separations between relativism and foundationism (the latter being, for our purposes, akin to realism). She disavows both by locating herself "inside" the acts of ordering and knowing. In her studies of Yoruba-speaking children in Southwestern Nigeria, she observes the messiness of experimentation and the energy she invests in putting things in place. This is not to imagine some ideal, cleaner, tidier science and knowledge making, but to understand 'realness as emergent' [31:37], always subject to and entangled in an unfolding assemblage of things being brought into relation to one another. Barad helps make sense of the apparatus *vis-à-vis* this 'relational achievement'. The apparatus, or the tools, are inseparable from the experiment, experimenter and phenomenal world, so that their "intra-action"—as Barad refers to it—are all assembled in what, exactly, the phenomenon is. "Every measurement involves a particular choice of apparatus, providing the conditions necessary to give meaning to a particular set of variables, at the exclusion of other essential variables" [2:115]. The apparatus are thus unavoidably implicated in what is observed (and not observed) and consequently what is known. Crucially, as Barad argues, this is not an anti-realist or relativist position. It is what she calls 'agential realism', a recognition that what is real is actively brought into—or performed in—the world. Through tool use, then, phenomena are understood to be in a continuous state of becoming.

Returning to BMA, these arguments offer an alternative conception of the proof-mode problem. They help to see tools using top-down structural modeling techniques such as BMA as new ways of interacting with and, in a manner of speaking, enacting biology, and thus 'seeing' the phenomena anew. If computational tools are thought not as lenses onto a static phenomenal world—a presupposed ground truth—but instead part of the assemblage of things that constitute the "intra-action" of the experiment, we begin to see how the proof-mode problem is not just an obstacle to be overcome by re-introducing some notion of 'real' or 'natural' processes, e.g., biological time. Rather, the problem becomes one of enabling one particular relational achievement over (or as well as) another. And, critically for design, the challenge becomes how to make such unfolding and emergent relational achievements intelligible so that one is able to 'move on'.

This, then, offers a way of seeing the frictions—apparently rooted in disciplinary and conceptual differences—as opportunities to open up the sequence of practices. Knowing isn't treated as passively waiting to be found in "objects-ready-made", as Verran refers to them [31:152], but contingent and situated in practice. The tool becomes integral to these unfolding enactments, not merely a better or worse lens, but an active agent in constituting what is known. With this, the frictions open up opportunities for representing multiple figurings of phenomena and consequently enabling biology to be enacted in divergent and new ways.

Appropriately, perhaps, the consequence of framing things in these terms on BMA's design is still being worked out by the team. However, as part of the tool's ongoing development, the possibility has led to quite a different approach to designing the output of the analysis. Our efforts have been to test a range of divergent representational forms, some potentially inconsistent. Furthermore, we have experimented with both incorporating a number of these into the tool's proof mode and giving the user a straightforward means of creating mini or sub-experiments. The former has been driven by a goal to offer multiple and in some cases divergent ways of working with and making sense of the model data at the cost of consistency. The latter is intended to encourage a succession of views of the analysis that can be easily compared and contrasted, and that help users piece together a picture of the interventions they've made.
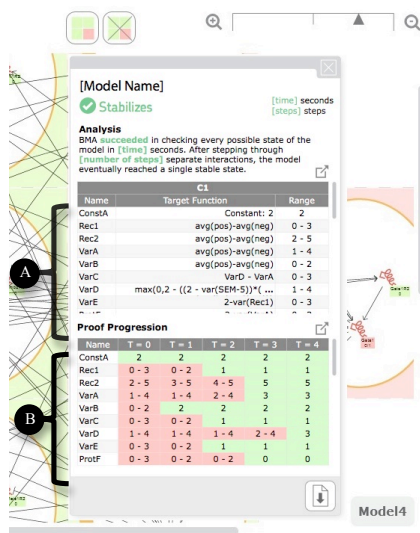

Figure 4. Mock-up of proof mode and output to analysis.

An early mock-up for presenting the results in the UI (Fig. 4) offers an indication of what this might look like. The interface is probably overly formal for one meant to promote open-ended thinking, but the general goal has been to give the user a way of seeing the proof's results differently, allowing them to shift views and easily intervene in the analyses. In the top table (A), the user can edit the network's variables (i.e., functions, ranges, etc.) and the proof is automatically re-executed. As well as the graphical representation of the network being updated, a "Proof Progression" table is displayed (B), representing how the variables' ranges change over a sequence of stages to reach stability. Thus, proof time is visualized in such a way that the graphical/structural and temporal representations are set against each other, encouraging users to try out modifications.

Not shown here are two other additions to the UI that have been conceived of in the same vein. One introduces a means of identifying and graphically representing nodes that are 'key' in the progression of the proof (using *directed acyclic graphs*). These graphs give users an idea of what if

any variables are key to the proof's analysis and what modifications may introduce significant changes. A second is a table listing changes made to variables as mini-experiments so that users can quickly switch between different network configurations and make comparisons. Again, the overall intention is to construct a UI for the proof mode that shows different representations of the analysis, but to make this intelligible and as manageable for the user as possible.

IMPLICATIONS AND CONCLUSIONS

On the face of it, this paper has a straightforward message, one showing how a grounded investigation into the practices of computational biology can help with designing constructive tools for biologists. From one viewpoint, we've shown that seeing tools like BMA as part of a set of practices for getting on with scientific work allows apparent problems arising from the use of computational techniques to be constructively refigured. From a second viewpoint, we've shown how UI design might be approached to support this notion of science as a sequence of entangled practices. We've found that opening up the interface to incorporate multiple representational forms resonates with the continuous and unfolding interventions scientists perform in their routine work. In this light, the paper provides an illustrative example of the work at the intersection of computation and biology, the kinds of problems that can be faced, and how in at least one case they might be tackled.

There is, however, a wider message we hope to have conveyed. This is more to do with how HCI approaches building tools in areas where computation is changing the ways scientific questions are being asked and answered. It is worth reiterating, such tools are establishing themselves across the gamut of research from the natural sciences to the social sciences and humanities, as well as into the more familiar domain of social networking/analysis. Seen in terms of pragmatic on the ground work, design in these areas opens up significant opportunities. Room is made for different ways of knowing when it is understood that, to borrow Barad's words, "representations are not (more or less faithful) pictures of what is, but productive evocations, provocations, and generative material articulations or re-configurations of what is and what is possible" [2:389]. Thus, design has a part to play in showing knowledge to be something underway, and the role tools have in intervening in these processes. BMA hopefully illustrates, for example, how design might be used to resist conformity to a single perspective. This is not just an idea of design that promotes different views onto the same phenomena. It locates design and HCI within an orientation that contests science as a monistic endeavor and supports instead an emergent idea of scientific ways of knowing.

Finally, something that has remained implicit thus far are the motivations behind and implications of an approach that prioritizes an 'entangled co-emergence' of agencies and knowings [2]. By accepting that the phenomenal world out-there is, so to speak, enacted—not merely discovered—we

are forced to face the prospect of designing tools that provoke very particular questions and trajectories. As the science and technology studies scholar John Law reasons:

> "[i]f out-thereneses are constructed or enacted rather than sitting out there waiting to be discovered, then it follows that their truth or otherwise is only one of the criteria relevant to their creation... But the moment we acknowledge this we are faced with new questions. What kind of out-thereneses are possible? Which are so embedded that they cannot be undone? Where might we try to undo or redo them? How might we try to nudge research programmes in one direction rather than another?" [21:40]

Thus, in a modest way, what we have presented is motivated by a need to extend the work at the interface beyond the immediate problems of representation. It is driven by a determination to acknowledge and begin taking seriously HCI's role in ways of knowing. With BMA this position has led us, tentatively, to explore ways of revealing the multiple agencies that are assembled to enact biology. The wider programmatic implications for HCI though are to do with the role the field might take in designing tools that offer new ways of both asking and answering questions. Our work hopefully animates a discussion about the knowings we wish to promote in designing these tools, and how we should actively participate in shaping the processes of discovery and invention in science and technology.

ACKNOWLEDGMENTS

REFERENCES

1. Amann, K. and Knorr Cetina, K. The fixation of (visual) evidence. *Human Studies 11*, 2 (1988), 133–169.
2. Barad, K.M. *Meeting The Universe Halfway*. Duke Univ. Press, 2007.
3. Benque, D., Bourton, S., Cockerton, C., et al. Bio Model Analyzer: Visual tool for modeling and analysis of biological networks. In *Proc. CAV '12*, Springer-Verlag (2012). 686–692.
4. Bietz, M. J., Ferro, T. and Lee, C. P. Sustaining the development of cyberinfrastructure: an organization adapting to change. *CSCW '12*, ACM Press (2012), 901-910.
5. Bietz, M. J. and Lee, C. P. Collaboration in metagenomics: Sequence databases and the organization of scientific work. *ECSCW '09*, Springer (2009), 243-262.
6. Bowker, G. and Star, S. L. *Sorting Things Out*. MIT Press, Cambridge, 1999.
7. Carlson, R.H. *Biology is Technology*. Harvard Univ. Press, Cambridge MA, 2011.
8. Chabrier, N. and Fages, F. Symbolic model checking of biochemical networks. *CMSB '03*, Springer-Verlag (2003), 149–162.
9. Cook, B., Fisher, J., Krepska, E., and Piterman, N. Proving stabilization of biological systems. *VMCAI '11,*
10. Cook, B., Podelski, A., and Rybalchenko, A. Proving program termination. *Comms ACM 54*, 5 (2011), 88-98.
11. Fisher, J. and Henzinger, T.A. Executable Cell Biology. *Nature Biotechnology 25*, 11 (2007), 1239–1249.
12. Fisher, J., Harel, D., and Henzinger, T.A. Biology as reactivity. *Comms ACM 54*, 10 (2011), 72–82.
13. Funahashi, A., Morohashi, M., and Kitano, H. CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. *Biosilico 1*, 5 (2003), 159–162.
14. Giere, R.N. Is computer simulation changing the face of experimentation? *Phil. Studies 143*, 1 (2009), 59–62.
15. Hacking, I. *Representing and Intervening*. Cambridge Univ. Press, Cambridge UK, 1983.
16. Ideker, T., Galitski, T., and Hood, L. A new approach to decoding life: systems biology. *Annu Rev Genomics Hum Genet 2* (2001), 343–372.
17. Keller, E.F. Models of and models for: Theory and practice in contemporary biology. *Philosophy of Science 67*, (2000), S72–S86.
18. Keller, E.F. *Making Sense of Life*. Harvard Univ. Press, London, 2002.
19. Krohs, U. and Callebaut, W. Data without models merging with models without data. In F.C. Boogerd, ed., *Systems biology: Philosophical foundations*. Elsevier, Amsterdam, 2007, 181–283.
20. Latour, B. The whole is always smaller than its parts: A digital test of Gabriel Tarde's monads. *British Journal of Sociology*, (in press).
21. Law, J. *After Method*. Routledge, London, 2004.
22. Leahu, L. Representation without representationalism (Doctoral dissertation). Cornell Univ., Ithaca, NY, 2012.
23. Myers, N. Conjuring Machinic Life. *Spontaneous Generations 2*, 1 (2009), 112–121.
24. Pickering, A. *The Mangle of Practice*. University of Chicago Press, Chicago IL, 1995.
25. Rheinberger, H.J. *Toward a History of Epistemic Things*. Stanford Univ. Press, Stanford, CA, 1997.
26. Rosner, D.K. The material practices of collaboration. *CSCW '12*, ACM Press (2012), 1155–1164.
27. Suchman, L. Agencies in technology design: Feminist reconfigurations. *Workshop on Gendered Innovations in Science and Engineering*, Stanford Univ. (15 April, 2005).
28. Suchman, L. *Human–Machine Reconfigurations: Plans and Situated Actions, 2nd Edition*. Cambridge Univ. Press, Cambridge UK, 2006.
29. Taylor, A.S. Machine intelligence. *CHI '09*, ACM Press (2009), 2109–2118.
30. Taylor, A.S. Out There. *CHI '11*, ACM Press (2011), 685–694.
31. Verran, H. *Science and an African logic*. Univ. of Chicago Press, Chicago IL, 2001.
32. Wiberg, M., Ishii, H., Dourish, P., et al. "Material interactions": from atoms & bits to entangled practices. *EA-CHI '12*, ACM Press (2012), 1147–1150.