Biol Theory (2015) 10:50–59
DOI 10.1007/s13752-014-0198-3

THEMATIC SECTION ARTICLE: EVOLUTIONARY SYSTEMS BIOLOGY

Engineering and Biology: Counsel for a Continued Relationship

Brett Calcott · Arnon Levy · Mark L. Siegal · Orkun S. Soyer · Andreas Wagner

Received: 23 May 2014 / Accepted: 29 November 2014 / Published online: 14 January 2015
© Konrad Lorenz Institute for Evolution and Cognition Research 2014

Abstract Biologists frequently draw on ideas and terminology from engineering. Evolutionary systems biology—with its circuits, switches, and signal processing—is no exception. In parallel with the frequent links drawn between biology and engineering, there is ongoing criticism against this cross-fertilization, using the argument that over-simplistic metaphors from engineering are likely to mislead us as engineering is fundamentally different from biology. In this article, we clarify and reconfigure the link between biology and engineering, presenting it in a more favorable light. We do so by, first, arguing that critics operate with a narrow and incorrect notion of how engineering actually works, and of what the reliance on ideas from engineering entails. Second, we diagnose and diffuse one significant source of concern about appeals to engineering, namely that they are inherently and problematically metaphorical. We suggest that there is plenty of fertile ground left for a continued, healthy relationship between engineering and biology.

Keywords Adaptationism · Design · Engineering · Evolvability · Gene regulation · Metaphor · Evolutionary systems biology

B. Calcott (✉)
School of Life Sciences, Arizona State University, Tempe, AZ, USA
e-mail: brett.calcott@gmail.com

A. Levy
Hebrew University of Jerusalem, Jerusalem, Israel

M. L. Siegal
Department of Biology, Center for Genomics and Systems Biology, New York University, New York, NY, USA

O. S. Soyer
School of Life Sciences, University of Warwick, Coventry, UK

A. Wagner
Institute of Evolutionary Biology and Environmental Studies, University of Zurich, Zurich, Switzerland

A. Wagner
Swiss Institute of Bioinformatics, Lausanne, Switzerland

A. Wagner
Santa Fe Institute, Santa Fe, NM, USA

Introduction

Biologists draw on engineering as a matter of course. Examples include the idea that cells and proteins can be viewed as computational devices (Bray 1995), that concepts from engineering such as integral feedback control can be used to describe regulatory interactions (Csete and Doyle 2002), and that such systems are analogous to digital logic circuits (Kauffman 1969). Criticisms of appeals to engineering are also widespread (Lewontin 1996; Pigliucci and Boudry 2011; Nicholson 2012; Boudry and Pigliucci 2013). Such criticisms highlight differences between organisms and engineered devices, urging caution in attempts at cross-fertilization or recommending that biologists cease to rely on concepts and tools from engineering.

Here we aim to clarify and reconfigure the link between biology and engineering, presenting it in a favorable light. We do so by, first, arguing that critics operate with a narrow and misleading notion of how engineering actually works and of what the reliance on ideas from engineering entails. Second, we diagnose and diffuse one significant source of concern about appeals to engineering, namely that they are inherently and problematically metaphorical. To be clear: our goal is not to suggest that an engineering

perspective is always a useful approach to thinking about biology. Rather, it is to show that engineering has a richer set of resources to draw on than it is typically credited with, and that drawing on these resources involves more than loose comparisons and suggestive imagery.

We begin by outlining the critical stance towards viewing biological systems in terms of engineering and design, focusing on François Jacob's (1977) classic discussion. Jacob's central claim is that the process of evolution differs in key respects from the process of design and construction in engineering, so the "artifacts" it generates are very different from those produced by engineers. We concede that Jacob's claim looks reasonable when evolution is compared to an idealized image of how engineering works. We argue, however, that the claim looks far less plausible when a broader perspective on engineering is taken, especially when we move beyond familiar mechanical examples and look at disciplines such as software engineering. Furthermore, we argue that principles and methods from engineering may prove useful to biologists even when the processes through which they come about are different.

In the second part of the article we discuss models and metaphors. Appeals to engineering are often regarded as metaphorical and this is one important source for hostility towards them. We aim to alleviate this concern by situating metaphors with respect to models, showing them to be species of a common genus, namely surrogative representation, where one thing is used to think about another thing. This allows us to suggest a more nuanced outlook on the interaction between engineering and biology. We argue that many of the worries associated with metaphors apply equally to models. The key difference, we suggest, is that the content of models is more precisely specified, and that one route engineering metaphors may take is to be gradually replaced or accompanied by more precisely specified models. Indeed, we think there is ample evidence of this: the metaphorical language deployed by biologists is often accompanied by models and analytical tools imported from engineering.

The examples we use throughout are related to evolutionary systems biology, because our ideas are developed primarily in the context of this field. But the basic claims we argue for apply to other parts of biology as well. One interesting case that we do not discuss is synthetic biology: the attempt to artificially construct cells, organisms, and other biological structures. This field involves a mix of engineering aims and already evolved biological parts. Although the issues it raises are related to what we say here, we prefer to leave these connections for a later discussion, and focus our attention here on cases where there is a clear distinction between evolved and engineered systems.

Engineering: Not Just About Engines

> … the systematic application of engineering metaphors to a domain that is fundamentally different from the world of human artifacts may send scientists on a wild goose chase. (Boudry and Pigliucci 2013, p. 667)
>
> … engineering has a long-standing status problem, best summed up by the greeting: "If you're an engineer, I've got a lawnmower that needs fixing." (Macilwain 2010, p. 885)

A familiar image of what engineering is and how engineers work recurs in discussions of engineering talk in biology. In this image, engineered artifacts consist largely of concrete mechanical or electrical devices (cars and planes are popular choices), and the process of design features a well-informed engineer at a drawing board conjuring up blueprints for a polished and optimized product that "merely" needs to be built. Perhaps this image captures some cases of engineering, but it fails to reflect the breadth and diversity of the field. Importantly, it neglects parts of engineering that are highly pertinent to biology.

In this section, we do three things. First, we separate criticisms of engineering products from criticisms of engineering processes. These are often run together in the standard view, and separating them allows us to evaluate them independently. Second, we use some examples from software engineering to demonstrate the shortcomings of the familiar image of engineering, and its tendency to overstate differences (and understate similarities) between engineering and biology. Lastly, we show how a broader view of engineering and of biology suggests some useful, but less familiar, ways of connecting the two areas.

Separating Process and Product

Two related concerns typically arise in connection with comparisons between biology and engineering. The first focuses on the *design process*—on how the relevant systems come into being and on whether this process is sufficiently similar to biological evolution. The second focuses on the *resulting product*—on the character of engineered systems versus evolved ones.

Both themes appear in what is perhaps the best-known modern text in this area: François Jacob's "Evolution and Tinkering" (Jacob 1977). As Jacob has it:

1. The engineer works according to a preconceived plan, foreseeing what will be produced.
2. The engineer starts with specially prepared materials and tools designed for the task at hand.
3. A good engineer produces something close to perfect, given the technology of the time.

Jacob contrasts these claims about engineering with the process and products of evolution. Evolution doesn't work to a plan, needn't start with the right materials for the job, and rarely produces things that are close to perfection. These contrasts lead Jacob to conclude that the analogy between engineering and evolution is a poor one.

Two central themes animate Jacob's discussion. First, there is a contrast between the *design processes*—engineers design on the basis of explicit goals and prepared materials, while evolution does not. Second, the *resulting products* of engineering are "close to perfect," while those produced by evolution are suboptimal and messy. Jacob links these two claims: the products of evolution are messy and imperfect *because* the evolutionary process is neither prepared nor goal-directed. Similar links between product and process can be found in many discussions of evolutionary change, though sometimes the argument runs in the other direction—from product to process. For example, Paley argued from the perfection of biological products to the existence of a goal-directed producer, namely God (Paley 1817). Adaptationists also argue from product to process, suggesting that optimality of the products is proof of the power of natural selection.

In general then, there is a cluster of ideas, sometimes explicitly stated but more often tacit, that surrounds the connection between product and design process in these debates. This strong connection can be misleading, for it is sometimes presumed that any claim about similarities between the products of evolution and engineering must invoke a claim about similarities in the processes responsible for them. For example, a frequent assumption is that any mention of "design principles" (or just "design") to describe an evolved structure implicitly relies on strong assumptions about the power of selection. This inference is coupled to a particular view of "reverse engineering," where to reverse engineer some artifact involves making a claim about its evolutionary history (Griffiths 1996; Lewens 2005). But if we look at a field such as biomimetics—where engineers produce designs inspired by a natural system—we find a version of reverse engineering and talk of "design principles" that is not coupled to evolutionary history (Calcott 2014; Green et al. 2014a). The "principles" that both engineers and biologists speak of in this realm refer only to the components and organization that endow organisms with particular capacities—how a gecko sticks to the ceiling, or what makes spider silk light and strong. These principles say nothing about an organism's past, or what selective advantage such features provide now. This should be evident, for the engineers interested in biomemetics *do not care about these issues.* They just want to know how things work, so they can copy them.

Consider the work on motifs in gene regulatory networks—a seminal piece of systems biology—in light of this separation between product and process (Alon 2006). Network motifs are small subgraphs within a larger network, such as feedforward loops and other simple patterns, which occur frequently in biological networks.[1] They exhibit interesting functional dynamics, especially in the context of gene regulation. A straightforward engineering claim is that the "wiring" together of a number of genes can produce "circuits" capable of common, familiar signal processing tasks, such as low-pass filtering and pulse generation. This is an intriguing claim about the capacities of very simple gene networks. It is a further step to claim that these network motifs are adaptations, that their processing task is their evolutionary raison d'être. To be sure, such claims have been made (Alon 2003). But as several authors have argued, these claims are premature, since regulatory motifs may also arise through nonadaptive processes (Cordero and Hogeweg 2006; Lynch 2007). That said, such arguments do not affect the original claim about the *product* of evolution—motifs may still have certain capacities regardless of whether they are products of natural selection or random drift. Whether and when these capacities were fitness enhancing is something we are still largely in the dark about. But it is certainly possible to formulate and evaluate hypotheses about the present-day capacities of these gene networks without invoking claims about their adaptive history. More generally, engineering-inspired claims can be directed at extant biological systems, and evaluated with respect to how well they capture present-day features, irrespective of whether they are products of natural selection. These points are fairly straightforward, but they are worth keeping firmly in mind when evaluating engineering-inspired biological work.[2]

The foregoing points pertain to the status of engineering-based work that targets the "here and now." Our claim has been that it is a mistake to treat such work as necessarily beholden to facts or hypotheses about how the features in question arose. In many cases, however, biologists do wish to connect engineering claims about the product to the process of evolution. Above we mentioned adaptationist claims, but claims about evolvability make this connection too—suggesting that properties, such as modularity, affect the process of evolution. Here, we think that views of engineering like Jacob's simply miss the mark, for

[1] One may distinguish a representation that portrays a biological system as a network from the connections and interactions among the biological entities themselves. It is not obvious that biological systems *are* networks, or even what that would exactly mean. But we must bracket this interesting issue here.

[2] This does not imply that we advocate a strong dissociation between claims about proximate and ultimate causation. There is a lively debate on that issue (Laland et al. 2013; O'Malley and Soyer 2012; Steinacher and Soyer 2012; Calcott 2013). We are merely making the modest claim that hypotheses about current behavior do not *as such* presuppose assumptions about evolutionary origins.

they operate with a narrow and largely misleading view of how much of engineering works. In what follows, we sketch a more realistic picture of the design process and the implications it has for the products of engineering.

A Realistic View of the Design Process

Biological systems, unlike engineered systems, are not constructed by conscious, goal-directed, designers. It does not follow, however, that the process of evolution and the process of human design are completely different. If we examine the way some human engineers or inventors design artifacts, we find much in common with evolution. For example, trial and error play a crucial role in the work process of even the most visionary inventors. And while human engineers are certainly goal oriented in the short run and nature is not, their guesses as to what a new device might eventually be used for often fall flat. Edison, for instance, initially viewed his phonograph as primarily useful for business communication. Critics of the biology-engineering linkage often operate with idealized views of engineering that tend to overemphasize the conscious, planned capacities of designers, and neglect the iterative, error-prone process that actually takes place.

The recent history of software engineering provides an example of just how misleading this idealized version of design can be. Throughout the 1970s and 1980s, a widely used method of large-scale software design broke the process down into a series of independent steps, which could be executed sequentially (roughly: requirements and systems analysis, architectural design, programming, testing, and deployment). This view was dubbed the "waterfall model" (Royce 1970), and was adopted by such organizations as the US Department of Defense. Since its inception, the waterfall model has been criticized for its lack of feedback and iteration between various stages of software construction. Furthermore, adherence to this strict linear approach to design has often been identified as they key reason for the frequent failure of large-scale software projects, which some estimates put at over 50 % of projects (Ellis 2008). It is common, for example, for precise requirements to be unclear at the beginning of a project, and many complex design problems often don't come into view prior to implementation (McConnell 2004). Software projects that tried to follow the unidirectional, no-surprises waterfall model often encountered an explosion of problems late in the project, causing delays and project failures.

In the last 20 years, this process has largely been replaced with an iterative approach to software design, where feedback and crosstalk between construction and design pervade all stages of the process (though the "waterfall" idea still persists, most notably in management). This iterative approach also embraces a "release early, release often" maxim,[3] signifying a design process where successive iterations of software are quickly produced and fed back into subsequent design and production steps. This process has become increasingly important in an environment where the goals often change as quickly as software can be produced. In short: the design process, in some engineering disciplines, is explicitly iterative with testing and feedback, in view of imprecise and shifting goals. This is one key way in which some parts of engineering design differ from the traditional conception and, as we shall see, may bear closer similarities to biological evolution.

A related and crucial point is that this "loopy" process of engineering cannot and does not begin afresh at every iteration. For the most part, it consists of *updating and improving existing software*, in a manner familiar to anyone who owns a computer or a smartphone. In an update-based process, the materials for the job are typically inherited from previous design "generations"— previous versions of the software (indeed, it is possible to construct a "phylogeny" of these changes, where different versions fork or merge together). Moreover, possible changes are highly constrained by the existing design and by its compatibility with input formats and other elements of the preexisting "ecosystem." This results in a process of incremental and at times erratic modification. Thus, software changes in a manner that looks completely unlike the familiar image of engineering as the creation of a perfect blueprint from scratch, an image that dominates portrayals like Jacob's "Evolution and Tinkering."

Big Balls of Mud

If the standard image of the engineering process as a premeditated and perfectly rational process is deeply flawed, then so is the idealized view of the resulting product as being perfect, or nearly so. This idea of near-perfection is often used to contrast biology and engineering. One argumentative tactic is to identify some ostensibly "poor design" in biology, such as the back-to-front architecture of the vertebrate retina,[4] and then follow this with a suggestion that no engineer would produce anything as messy and needlessly complex. This contention loses much of its plausibility when we are faced with the actual practice of disciplines such as software engineering.

A more realistic portrayal is given by computer scientists Brian Foote and Joseph Yoder: they suggest that the

[3] See http://en.wikipedia.org/wiki/Release_early,_release_often for a summary of these ideas.

[4] It is worth pointing out that a charge of poor design is only possible if we have some standard of what good design is—so there is an implicit use of engineering in these arguments.

defacto standard for software architecture is a "Big Ball of Mud" —a "haphazardly structured, sprawling, sloppy, duct-tape and bailing wire, spaghetti code jungle" (Foote and Yoder 2000). Foote and Yoder note that the "overall structure of the system may never have been well-defined" or that it "may have eroded beyond recognition." They suggest a number of reasons why such architectures prevail, including constraints on time and cost, but also issues such as complexity, shifting goals, and problems of scaling—a big project isn't just the sum of multiple little projects put together. Thus, the structure of large complex systems that have been assembled by engineers can have many of the properties ordinarily thought to be unique to evolved systems, and for many of the same reasons: biological systems function under severe regimes of costs and constraints, environmental and other challenges shift, and even the simplest organisms are immensely complex. Why do engineers produce such terrible architecture? Foote and Yoder's reply is reminiscent of how a biologist would explain the "poor design" of the vertebrate eye: "People build Big Balls of Mud because they work" (Foote and Yoder 2000).

The overall lesson is that once we step outside a superficial and clichéd image of engineering, the "fundamental differences" between biology and engineering become less marked. Moreover, as we shall discuss next, positive insights can be gleaned from drawing better comparisons.

Engineering as Smart Tinkering

When Foote and Yoder explain why software can become so reticulate and haphazardly constructed, they reason much as Jacob did for biological systems: the messy nature of the product is a consequence of various constraints on the process, such as shifting goals—corresponding to changing selective environments in biology—and no ability to start afresh –which biologists refer to as historical contingency.[5] Viewed through examples like software design, engineering becomes far more like Jacob's tinkering than it resembles the standard view of engineering.

The next step in their article—to look for ways to avoid architectural disasters—embodies the consequences of this shift in focus. It requires thinking about the sorts of structures and practices that perform better when subject to continual iterative change. In other words, Foote and Yoder are asking *what structures are more apt for tinkering*. This question is closely analogous to a question asked by biologists: what makes biological systems evolvable? The answers highlighted by software designers mirror some of those in the biological literature on evolvability—modularity being the most obvious. This connection between engineering and biology—where a complex system's properties affect how the system changes over time rather than how it performs at any one time—remains largely unexplored.

One exception is Raman and Wagner's work on the "evolvability of digital circuits," which illustrates this connection (Raman and Wagner 2011). They explore a class of programmable digital circuits that compute logic functions, to examine whether the space of possible circuits contains "neutral networks"—sets of circuits that compute exactly the same function, with one being reachable from another via a small number of mutational steps. This property, where multiple different "genotypes" can encode the same "phenotype," forms a key part of Wagner's theory of evolutionary innovation, for it allows the robust exploration of a space of possibilities (Wagner 2011). These same properties also exist in the space of digital circuits, leading to the conclusion that "properties important for the evolvability of biological systems exist in a commercially important class of electronic circuitry" (Raman and Wagner 2011, p. 269).

It is worth noting that the idea of multiple equivalent ways of solving a problem is something foreign to the standard view of engineering, which envisages engineers as producing a unique and highly optimized solution (Jacob 1977). Yet it is often found in software engineering, where "refactoring"—modifying the architecture of the system without changing its function—is essential to software updating, suggesting that neutral networks are possible in software too (Calcott 2014).

In summary, when we take a fresh, and more realistic, look at engineering, we see that many criticisms are premised on a misconception of the way engineers actually come up with the products they design. The standard view of engineering not only fails to capture several similarities between engineered and evolved systems, it also obscures an important question: what structural properties make a system more amenable to tinkering?

Needless to say, we are not suggesting that engineering and biology are *identical*. But we do think that a more productive dialogue about the similarities between engineering and biology requires being clear about the biological context, and taking seriously the resources offered by the enormous variety of engineering disciplines that exist. For example, there are clear links between biomechanics and mechanical engineering. We have argued that issues in software engineering and questions about evolvability can also be linked. Other issues, such as assumptions about the decomposability or fragility of engineered systems in contrast to biological systems should also be revisited with an

[5] William Wimsatt's work on generative entrenchment has, for many years, emphasized similarities between evolutionary and technological change (Wimsatt 2007).

eye to comparing appropriately complex engineered systems. These links (or contrasts) must be made point by point with an eye on both the engineering details and the scope of biological application—there is no general answer to the question "Is biology like engineering?"

From Metaphors to Models

A recurring suspicion about biologists' appeals to engineering stems from viewing them as metaphorical (Boudry and Pigliucci 2013; Pauwels 2013). There is a tendency, in many discussions, to regard the very use of metaphors as problematic or unscientific or to think that the only way to deal with a troublesome metaphorical association is to eliminate it wholesale.[6] We think this is an overreaction: while metaphors may mislead us, they are not unique in this. And, more importantly, it is possible to "rein in" a metaphor—by turning it into a model. To clarify these claims and argue for them, we now situate metaphor, alongside models, within the broader class of surrogative representation.

In surrogative representation a speaker or writer regards one thing—one's *target system*—as if it were another. Doing so can highlight specific aspects of the target and may permit some insight by moving to a simpler or more well understood case. The use of metaphor in science is often characterized this way (see, for example, Pigliucci and Boudry 2011, p. 465), and modeling in science works much the same way (Godfrey-Smith 2006; Weisberg 2007; Levy 2014a, b). When we model a biological system of regulatory interactions as a Boolean network (Kauffman 1969; Thieffry and Romero 1999) we effectively treat the target object—a regulatory system—as if it were a different thing: a set of logic gates connected by switches. Thinking in terms of a simplified representation the ultimate target affords a variety of epistemic advantages, such as tractability, highlighting certain effects or factors, thereby facilitating the communication of ideas and results. Metaphors and models are thus members of a broader family—they are both forms of a common cognitive strategy of surrogative representation. The purpose of the relationship between the "vehicle"—a metaphor or model—and the "target" of representation can vary: it might be used to explain, or to predict, or to explore. The standards for evaluating the success of a metaphor or model vary accordingly.

While metaphors and models share this much, they differ in some important respects. We will highlight two such respects: specification and matching. Specification concerns the way in which the vehicle is characterized—the degree of clarity and precision with which the model or metaphor is formulated. In contrast, matching pertains to the relationship between the vehicle and the target—whether and how it corresponds to the properties of the target, given the task at hand.[7]

Specification

There is a substantial difference in how explicitly and clearly models and metaphors are *specified*. In metaphor, the vehicle itself is not typically described in detail, which of its properties are most relevant is often unclear, and exactly how the similarity between the vehicle and target is meant to work is left open. The upshot is that it is difficult to know whether two people understand the connection in the same way, as their interpretations may be influenced by their particular preconceptions or other idiosyncratic factors. A model, in contrast, is typically outlined in precise detail. Its content can be readily discerned and, most importantly, agreed upon by different researchers. This is central, as it affects the degree to which a surrogative representation can be assessed and deployed by a collective, interpersonal body such as a scientific community.

We can illustrate these ideas with two examples whose target object is DNA. First, consider the notion that genetic material resembles a text. Perhaps the best-known example is the tendency—now somewhat less common than it was a decade ago—to describe the genome as the "book of life." What exactly follows from describing genetic material as text-like, or as a book? Does it contain analogues of words, sentences, or chapters? Does it have a beginning and an end? Should we understand the metaphor to mean that knowledge of the "language" in which the book is written is sufficient (or nearly so) for understanding the ins and outs of inheritance and development? It seems that no substantial agreement exists (nor has one ever existed) on the answers. To be sure, we have some idea of how to interpret the book metaphor—it directs our thinking in some ways, for example, towards a primary linear structure, and towards read/write mechanisms—and so it has certainly served to inspire research and guide our thinking about the genome. But these roles do not require a resolution of the ambiguities involved.

In contrast, consider a worm-like chain model—a standard simple tool for studying the mechanics and spatial

[6] "We…suggest that biological research and teaching could and should actually be done without much use of metaphorical thinking…" (Pigliucci and Boudry 2011, p. 455).

[7] There are other forms of surrogative reasoning that share some properties with metaphor and some with models. A significant example is analogy. We leave it to the reader to extrapolate from what we say here to other cases of surrogation, since our aim is not to cover this topic in exhaustive detail.

organization of polymers such as DNA in solution. Despite the colorful name, this is not a mere metaphor. Here a multi-unit polymer is treated as if it were a long, uniformly flexible rod. Such a model is often used to assess, in quantitative terms, the extensibility of a DNA molecule, the amount of force it can withstand, and related properties (Rubinstein and Colby 2003; Nelson et al. 2013). In analyzing the worm-like chain model, it is clear what the model says, how it depicts its target, and what the implications of this description are. In this case, unlike the book-of-life metaphor, it is evident that questions about the content of the model—what it says, how it describes the world—can be evaluated by different persons in a shared and uncontroversial manner.[8]

There are many cases that are in between these extremes, where a theoretical surrogate is somewhat open-ended: not to the extent of a metaphor such as the genetic book of life, but also not having a precision and objectivity akin to the worm-like chain model of DNA polymers. Metaphors and models can be thought of as varying along a spectrum of surrogative representation, differing in terms of clarity and precision. Metaphors lie at the "opaque" end of this specification spectrum, while models sit at the "transparent" end.[9]

For some purposes, and in some phases of a research project, the lack of precise specification characteristic of a metaphor may have utility. The open-endedness of a term and lack of definite content (whether metaphorical or not) may benefit a research community, because it allows its members to work under a common umbrella, while smoothing out differences in how exactly the relevant research agenda, assumptions and implications, are understood. Furthermore, a theoretical construct's position along the specification spectrum is not a static matter. An idea that originated as a metaphor can get transformed into a model (and vice versa). Often, this occurs when the metaphor serves as the basis for a mathematical formulation. Even when a mathematical treatment is not feasible, a metaphor may be made more precise and transparent—more like a model—via verbal or graphical means. If a metaphor's capacity to evoke ideas and facilitate communication becomes outweighed by its tendency to let in subjective associations and thus mislead its users, then it may be time to "modelize" it. We return to this idea below.

[8] Note that precise specification makes the *content* of the model uncontroversial. Whether and how the model *matches* the biological system may remain controversial.

[9] As one reviewer suggested, a metaphor and a model might differ in other respects too, such as the status of vehicle of representation: a metaphor might be linguistic, while a model is often mathematical. We won't delve into thorny issues concerning linguistic versus other forms of representation here. It suffices for our purposes that precision is *one significant difference* between models and metaphors.

Matching

We have suggested that a key difference between models and metaphors is how explicitly they are specified. We now consider a second distinction, *matching*. Matching concerns the fidelity with which a model/metaphor depicts its biological target. This is where problems with metaphors are typically thought to lie: engineering metaphors are thought to be misleading because biological systems differ from engineered systems in important ways. Hence the utility of the vehicle as a means for representing and reasoning about the target is called into question.

Given that both models and metaphors are surrogative representations, how do they differ with respect to matching? A full discussion of matching is well beyond the scope of this paper, as this involves complex questions concerning representation and "aboutness" (Weisberg 2013 discusses this at length). But a number of key points concerning the differences among models and metaphors can be made here. We will examine this issue with the aid of the frequently used idea that gene regulation can be treated as a Boolean logic circuit. The core of this idea is that individual genes are like simple logic gates that compute Boolean logic functions—AND, OR, etc.—and that these genes are linked through regulatory connections that are analogous to the wiring between logic gates on a digital logic circuit (Balch 2003). The (on–off) expression state of a set of genes can thus be thought of as being computed by a network of interconnected logic gates. Mutations can be thought of as changing either the logic function computed by a gate or the interconnections among gates in various ways.

What can we say about this idea as a metaphor? There are many differences between gene regulation and Boolean circuits. The regulatory systems inside cells are highly noisy, on a level that would not be tolerable in electronic systems—there are even cases where this noise helps the regulatory system perform its function (Eldar and Elowitz 2010). Furthermore, regulatory systems operate in a continuous space, because levels of gene expression are fundamentally continuous variables. In contrast, Boolean logic circuits perform computations in a discrete manner. If these sorts of differences are at the forefront of our interpretation of the circuit metaphor, then we're likely to think that Boolean logic gates are misleading representations of gene regulation (Pauwels 2013). But it is often hard to tell whether general metaphorical talk of gene regulatory circuits does indeed carry such implications.

Often the idea of a genetic regulatory circuit functions not as a vague metaphor, but as an explicit model of a particular biological system—a precisely formulated network making well-specified predictions. In this case the content of the surrogative vehicle is clearer, hence we

regard it as a model. But the match with the target may still be highly limited. Boolean circuits may be poor representations of regulatory systems, for example if noisiness or continuous gene expression states are important for the function under investigation. This observation illustrates that the greater transparency of models as surrogative representation does not imply that there is a good match between the model and the target system. The transparency of a model might make it easier to assess the vehicle-to-target match. But it does not, in and of itself, make a difference to how good that match is.

So here is a key point: models, just like metaphors, can mislead us, and thus *poor representation is not solely the province of metaphors*. Models can mislead us for reasons similar to those that apply to metaphors—for the simplifications they employ may lead to significant failures in matching the relevant biological phenomena. Put differently, the whole point of metaphors and models—to highlight or emphasize particular features—is the reason why any such surrogate representation can mislead us; it will never perfectly match its target.

So how are we to judge whether the match is a good one or not? We don't think there are any hard and fast generalizations to be had here. Matching is a complex and context-specific matter. To judge whether or not a match is good, we need to look not just at the degree of resemblance between vehicle and target, but also at the research task at hand, and the degree and kind of correspondence to the world it mandates.

Consider, for example, two very different uses of Boolean circuits to model gene regulation. The first is by Peter and Davidson (2011). They construct a Boolean model of gene regulation that captures the core regulatory interactions of a well-studied system—the early development of the sea urchin embryo. The model not only reproduces a large variety of spatial and temporal gene expression patterns, but perturbations to the model also reproduce changes observed experimentally. The central aim of this model was to represent the experimental observations of a particular regulatory system, and it appears to do this very well.

The second model is far more abstract, and comes from work on modularity and evolvability by Kashtan and Alon (2005). They simulate the evolution of a simple Boolean circuit under a number of different selective regimes in silico, and demonstrate that the model gives rise to a modular architecture when the environment rapidly switches between two structured challenges. The object of study here, rather than being a single system, is a far broader target: "modularity" in general. Furthermore, the model includes selective components and the evolutionary trajectory over many generations rather than single perturbations. The model is significant since, as Gunter Wagner points out, it is one of the few models of the evolution of modularity that we have (Wagner et al. 2007).

The breadth of Kashtan and Alon's model, and a lack of close mapping to a particular concrete regulatory system like that of the sea urchin makes it more difficult to assess how well this model captures biological reality. This less precise matching might be thought detrimental to the model. It is, however, an essential feature, as the model is being used to explore the *general* principles that underlie the evolution of modularity, rather than reproducing the phenomena seen in a *particular* empirical example. We see, then, that it is not possible to assess whether the mapping between model and target is doing a good job without taking into account the goals of the modelers (Weisberg 2013). This point applies to any kind of surrogative reasoning, and underlines the importance of precisely describing the intended aims of a model or a metaphor.

This suggests one important role that comparisons between engineering and biology can play: they force us to identify general principles that are applicable across a range of systems, rather than focusing on the specifics of a particular biological example (see Green et al. 2014b, this issue). This kind of abstraction is not always desired (because sometimes the details matter!), but seeking unifying principles is certainly one goal in science.

Towards a Healthy Marriage

We have argued that models and metaphors share important features, that they differ in degree rather than in kind, and that a particular idea may be presented and pursued in a more or less metaphorical (or model-like) way and may change its location along the model-to-metaphor spectrum. One lesson to draw from this is that we should be cautious in deciding whether an appeal to engineering is "merely" metaphorical—and, especially, in disparaging it for this reason. For example, the notion of a "pump" might sometimes function as a simple mechanical metaphor to think about biological systems that move "stuff" around (Pigliucci and Boudry 2011). But a cursory look at a journal article such as "The Sponge Pump: The Role of Current Induced Flow in the Design of the Sponge Body" (Leys et al. 2011) shows that "pump" can invoke a sophisticated mathematical model that deals in the same physical laws, constants, and problems as an engineer works with.[10]

[10] As Steven Vogel would have it: "… biomechanics has mainly been the study of how nature does what engineers have shown to be possible. Nature may have gotten there first, but human engineers, not biologists, have provided us with both analytical tools and practical examples" (Vogel 2003, p. 11).

Another lesson is that a concept or principle that functions metaphorically can often be transformed into a more model-like representation. One good reason to attempt such a transformation is that the metaphor, because of its relative looseness, allows misleading misconceptions to sneak into our conception of the relevant domain. We believe this is the best way to construe the worries about metaphorical appeals to engineering: it is not the fact that they involve metaphor per se that is troublesome, but the risk that leeway in interpreting these metaphors, coupled to preexisting conceptions about what engineered artifacts are like and how engineers work, will lead biologists astray.

We think that the points we've made allow us to accommodate these concerns, while not doing away with the biology-engineering link. The keys to maintaining the link are, first, an abandonment of the clichéd image of the engineer as a start-from-scratch, think-ahead optimizer in favor of the smart tinkerer in the image of a software engineer. We make room for far more interesting comparisons to biology when we do not presuppose that all engineering is like (the cliché of) mechanical engineering.

Second, where metaphors open the door to unwanted preconceptions, we might do well to be more precise both about the aspects of engineering that are being drawn upon and about the biological target domain to which they are meant to apply. Eventually, the goal would be to arrive at precise and well-matched models of the same phenomena. We can begin to flesh this out by recalling some of the earlier discussion. One source of trouble is the conflation of claims about product with claims about process. As we saw, in cases such as network motif work this has been relatively clear: a network motif model describes what a gene network does here-and-now, and that may have nothing to do with its evolutionary history or its adaptive value.

Lastly, principles from engineering disciplines that focus on modifying preexisting products, like software engineering, have the potential to illuminate change in biological systems. Modularity, for example, is a central theme in modern software development, as it has been in much recent work on evolvability. In both cases, modularity is meant to make a system more apt for tinkering. We suspect that there are substantial insights to be gained from the history of attempts—both successful and failed—to build large, scalable, rapidly changing modular software, insights that might clarify how modularity works, what it contributes, and how it arises in biology.

Like many complex relationships, engineering and biology can steer a course between an idealized unity and an ugly divorce. We wish them a healthy, reflective, and considered future together.

References

Alon U (2003) Biological networks: the tinkerer as an engineer. Science 301:1866–1867

Alon U (2006) An introduction to systems biology. Chapman & Hall/CRC, Boca Raton

Balch M (2003) Complete digital design. McGraw Hill Professional, New York

Boudry M, Pigliucci M (2013) The mismeasure of machine: synthetic biology and the trouble with engineering metaphors. Stud Hist Philos Sci Part C 44:660–668

Bray D (1995) Protein molecules as computational elements in living cells. Nature 376:307–312

Calcott B (2013) Why how and why aren't enough: more problems with Mayr's proximate-ultimate distinction. Biol Philos 28:767–780. doi:10.1007/s10539-013-9367-1

Calcott B (2014) Engineering and evolvability. Biol Philos 29:293–313

Cordero OX, Hogeweg P (2006) Feed-forward loop circuits as a side effect of genome evolution. Mol Biol Evol 23:1931–1936. doi:10.1093/molbev/msl060

Csete ME, Doyle JC (2002) Reverse engineering of biological complexity. Science 295:1664–1669

Eldar A, Elowitz MB (2010) Functional roles for noise in genetic circuits. Nature 467:167–173

Ellis K (2008) Business analysis benchmark: the impact of business requirements on the success of technology projects. IAG Consulting. http://www.iag.biz/resources/library/business-analysis-benchmark.html. Accessed 9 Dec 2014

Foote B, Yoder J (2000) Big ball of mud. Online version at http://www.laputan.org/mud/. Accessed 9 Dec 2014

Godfrey-Smith P (2006) The strategy of model-based science. Biol Philos 21:725–740

Green S, Fagan M, Jaeger J (2014a) Explanatory integration challenges in evolutionary systems biology. Biol Theory. doi:10.1007/s13752-014-0185-8

Green S, Levy A, Bechtel W (2014b) Design without adaptationism. Eur J Philos Sci. doi:10.1007/s13194-014-0096-3

Griffiths PE (1996) The historical turn in the study of adaptation. Br J Philos Sci 47:511–532

Jacob F (1977) Evolution and tinkering. Science 196:1161–1166

Kashtan N, Alon U (2005) Spontaneous evolution of modularity and network motifs. Proc Nat Acad Sci USA 102:13773–13778

Kauffman SA (1969) Metabolic stability and epigenesis in randomly constructed genetic nets. J Theor Biol 22:437–467

Laland KN, Odling-Smee J, Hoppitt W, Uller T (2013) More on how and why: cause and effect in biology revisited. Biol Philos 28:719–745. doi:10.1007/s10539-012-9335-1

Levy A (2014a) Modeling without models. Philos Stud. doi:10.1007/s11098-014-0333-9

Levy A (2014b) Machine-likeness and explanation by decomposition. Philos Impr 14(6):1–15

Lewens T (2005) Organisms and artifacts. MIT Press, Cambridge

Lewontin RC (1996) Evolution as engineering: integrative approaches to molecular biology. MIT Press, Cambridge

Leys SP, Yahel G, Reidenbach MA et al (2011) The sponge pump: the role of current induced flow in the design of the sponge body plan. PLoS ONE 6:e27787

Lynch M (2007) The evolution of genetic networks by non-adaptive processes. Nat Rev Genet 8:803–813

Macilwain C (2010) Scientists versus engineers: this time it's financial. Nature 467:885

McConnell S (2004) Code complete: a practical handbook of software construction, 2nd edn. Microsoft Press, Redmond

Nelson MR, King JR, Jensen OE (2013) Buckling of a growing tissue and the emergence of two-dimensional patterns. Math Biosci. doi:10.1016/j.mbs.2013.09.008

Nicholson DJ (2012) The concept of mechanism in biology. Stud Hist Philos Sci Part C 43:152–163

O'Malley MA, Soyer OS (2012) The roles of integration in molecular systems biology. Stud Hist Philos Biol Biomed Sci 43:58–68. doi:10.1016/j.shpsc.2011.10.006

Paley W (1817) Natural theology; or evidences of the existence and attributes of the Deity. Mason, London

Pauwels E (2013) Communication: mind the metaphor. Nature 500:523–524

Peter IS, Davidson EH (2011) Evolution of gene regulatory networks controlling body plan development. Cell 144:970–985

Pigliucci M, Boudry M (2011) Why machine-information metaphors are bad for science and science education. Sci Educ 20:453–471

Raman K, Wagner A (2011) Evolvability and robustness in a complex signalling circuit. Mol BioSyst 7:1081–1092

Royce WW (1970) Managing the development of large software systems (vol. 26). Presented at the proceedings of IEEE WESCON, Los Angeles

Rubinstein M, Colby RH (2003) Polymer physics. Oxford University Press, Oxford

Steinacher A, Soyer OS (2012) Evolutionary principles underlying structure and response dynamics of cellular networks. Adv Exp Med Biol 751:225–247. doi:10.1007/978-1-4614-3567-9

Thieffry D, Romero D (1999) The modularity of biological regulatory networks. Biosystems 50:49–59

Vogel S (2003) Comparative biomechanics. Princeton University Press, Princeton

Wagner A (2011) The origins of evolutionary innovations: a theory of transformative change in living systems. Oxford University Press, New York

Wagner GP, Pavlicev M, Cheverud JM (2007) The road to modularity. Nat Rev Genet 8:921–931

Weisberg M (2007) Who is a modeler? Br J Philos Sci 58:207–233

Weisberg M (2013) Simulation and similarity. Oxford University Press, New York

Wimsatt WC (2007) Re-engineering philosophy for limited beings: piecewise approximations to reality. Harvard University Press, Cambridge