Architecture, constraints, and behavior

John C. Doyle[a,1] and Marie Csete[b,1]

[a]Control and Dynamical Systems, California Institute of Technology, Pasadena, CA 91125; and [b]Department of Anesthesiology, University of California, San Diego, CA 92103

Edited by Donald W. Pfaff, The Rockefeller University, New York, NY, and approved June 10, 2011 (received for review March 3, 2011)

This paper aims to bridge progress in neuroscience involving sophisticated quantitative analysis of behavior, including the use of robust control, with other relevant conceptual and theoretical frameworks from systems engineering, systems biology, and mathematics. Familiar and accessible case studies are used to illustrate concepts of robustness, organization, and architecture (modularity and protocols) that are central to understanding complex networks. These essential organizational features are hidden during normal function of a system but are fundamental for understanding the nature, design, and function of complex biologic and technologic systems.

complexity

Systems approaches to biology, medicine, engineering, and neuroscience face converging challenges, because modern science, technology, and culture create dauntingly complex but similar and overlapping problems in these domains. Our goal is to develop more integrated theory and methods applicable to all systems, including neuroscience, by concentrating on organizational principles of complex systems. Beyond scientific understanding of systems, practitioners want to avoid and fix network errors, failures, and fragilities. This practical necessity requires mechanistic and often domain-specific explanations, not vague generalities. Therefore, universal theories must facilitate the inclusion of domain mechanisms and details and manage rather than trivialize their complexity.

Here, we aim to put recent progress in both experimental and theoretical neuroscience (1–15) in the context of a shared conceptual and mathematical framework (7, 16–32) in which a main theme is that complexity is driven by robustness and not by minimal functionality. We will emphasize robustness and efficiency tradeoffs and constraints and the control systems that balance them, their highly organized architecture (16–18), and its resulting side effects and fragilities. A confounding commonality that we must both overcome and exploit is that the most robust and powerful mechanisms are also the most cryptic, hidden from introspection or simple investigation. These mechanisms can give rise to a host of illusions, errors, and confusion, but they are also the essential keys to reverse engineering hidden network complexity.

This paper is inspired by several complementary research themes in behavioral neurosciences. The work by Marder (1) systematically perturbs both experimental and math models of small circuits to explore robustness and fragility properties of neural hardware in mechanistic detail. In humans, cleverly constructed experiments to unmask the workings of the brain can elicit visual (2) and other (3) illusions, suggesting hidden, automatic subconscious functions (4–6). A theoretical framework consistent with other empirical observations treats the brain as an integrated, robust control system (7) in which components for sensing, communication, computation, simulation, and decision are useful primarily to the extent that they effect action (8–12). Each theme (1–12) provides a separate constraint on the system as a whole, and therefore, seemingly dissimilar viewpoints can prove complementary and synergistic.

Our initial focus is how circuit (1) and system (2, 3) fragilities are necessarily the consequence of (not merely consistent with) implementing robust controllers (7) in such circuits. If brains evolved for sensorimotor control and retain much of that evolved architecture, then the apparent distinctions between perceptual, cognitive, and motor processes may be another form of illusion (9), reinforcing the claim that robust control and adaptive feedback (7, 11) rather than more conventional serial signal processing might be more useful in interpreting neurophysiology data (9). This view also seems broadly consistent with the arguments from grounded cognition that modal simulations, bodily states, and situated action underlie not only motor control but cognition in general (12), including language (13). Furthermore, the myriad constraints involved in the evolution of circuit and network mechanisms efficiently implementing robust control are essential to explaining the resulting fragilities, which vary from largely benign illusions (2) to dangerous dysfunction (3, 4, 16, 33–37) to potential catastrophes (16, 34–40).

In parallel to its broadening application in neuroscience, control theory and technology have expanded widely into networked, distributed, nonlinear, and hybrid systems in engineering and systems biology (e.g., ref. 16 and references therein and refs. 19 and 20). All these systems are of potentially great but as yet unrealized relevance to neuroscience as a source of both metaphors and new mathematics. Unfortunately, there is little shared language and few popular expositions (41). Thus, our next focus is to more broadly relate the studies in refs. 1–12 with the studies in refs. 16–20 while minimizing math and technical details. Using familiar case studies, we aim for accessible and concrete treatment of concepts such as constraints, tradeoffs, and layered architectures. Here, layering is functional and not necessarily mapping directly onto brain anatomy or physical architecture. An important example of layering is between computer hardware and software, but additional layering is a ubiquitous and essential architectural feature in complex networks of all types.

Neuroscience and Robust Control

A recent claim (7) is that human motor control is better explained as a robust rather than optimal controller, an explanation with a long tradition in neuroscience. Controllers optimal, on average, to only additive noise can be arbitrarily fragile to other uncertainties (21), motivating the development of robust control theory (22–24). Robust control is risk-sensitive, optimizing worst case (rather than average or risk-neutral) performance to a variety of disturbances and perturbations. Robust control theory formalized and extended best practice in control engineering and coincided with a massive expansion into active control in robots, airplanes, automobiles, smart weapons, com-

This paper results from the Arthur M. Sackler Colloquium of the National Academy of Sciences, "Quantification of Behavior" held June 11–13, 2010, at the AAAS Building in Washington, DC. The complete program and audio files of most presentations are available on the NAS Web site at www.nasonline.org/quantification.

Author contributions: J.C.D. and M.C. wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. E-mail: doyle@cds.caltech.edu or mcsete@ucsd.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1103557108/-/DCSupplemental.

munication networks, etc. Therefore, robust control is now ubiquitous but hidden, evidenced largely by what does not happen, such as skids, stalls, crashes, missed targets, dropped packets, etc. Similarly, most CNS activities are hidden from conscious awareness (4–6), implementing the sensing, decision-making, and actuation necessary for robust control in complex environments.

Control theory makes strong predictions about how robust circuits must necessarily be implemented largely independent of the device technology, all perfectly consistent with observations in neural circuits (1). Such claims are easily checked by experts in math, but hopefully, they are intuitively plausible to neuroscientists generally. In particular, any natural parameterization of functional control circuits (e.g., lobster somatogastric ganglia) (1) is well-known to be large (high dimension), thin (even higher codimension), and nonconvex (24). If it were otherwise, engineering design would be much easier. As a simple analogy to explain these terms, consider a 2D piece of paper with lengths that are large by some measure sitting in a 3D square box of comparable lengths. The larger that these lengths are, the smaller that the fraction of volume that the paper will occupy in the box is. Therefore, the paper can be both large and thin as a fraction of the box volume. If the paper is bent or wrinkled, then it is also nonconvex within the box, because most straight lines between two different points on the paper will not remain in the paper.

An even simpler example is the set of words in most languages, which is large but vanishingly thin as a fraction of all possible meaningless sequences of letters. There are 9! = 362,880 different permuted sequences from just the nine distinct letters adeginorz, roughly the total number of English words, but only organized is a word. Humans would have some difficulty checking this claim, but computers do so easily (and make formidable Scrabble opponents). The set of English words is, thus, large and thin. The set of functional parameter values of any circuit will also typically be large but vanishingly thin and nonconvex in the set of all possible (mostly nonfunctional) circuits. This fact is largely independent of the notions of function, circuit, or parameter, provided that they are sufficiently complex and realistic. Much of engineering theory is devoted to constructing special (higher-dimensional, nonphysical, and abstract) parameter embeddings that are convex and thus, algorithmically searchable for robust and functional design values. This idea that robust systems are large but thin and nonconvex in the space of all systems is a theme discussed below.

Another general feature of control systems is hard limits on robustness and efficiency (20). If the brain as controller has evolved to control action, then conscious thought may be, in some sense, a late evolutionary addition or byproduct (6–12). Both robust control theory and experimental evidence suggest that complex internal dynamic models are needed to resolve ambiguities in noisy sensations as well as plan for uncertain action, all in an uncertain, perhaps hostile, environment. When these models are implemented in slow neural hardware, management of the resulting delays almost certainly requires a heavily layered organization, a concept central to the network architecture emphasized here. (Delay has little ill effect if there is truly no uncertainty, because then, open-loop control is adequate; however, this ideal is never seen in practice.)

Pain and reflexes illustrate the sophisticated interplay of central and peripheral control, and fast action receives priority. Fast, thick, myelinated, general purpose sensory fibers initiate withdrawal from painful stimuli, whereas slow, thin, specialized fibers provide delayed, but detailed, information about the source of pain. This pattern is seen throughout the organization of the nervous system, and throughout behavior, we see this mix of reflex (fast, automatic, hidden, and expensive) and reflect (slow and conscious), with reflex receiving priority in resources. The acquisition of skill (playing instruments, ball sports, chess, reading, etc.) involves shifting down into fast reflex processes that start high and slow. Indeed, the more expert that we are in an activity, the less that we necessarily rely on conscious processes to perform, and such evidence for layering is found everywhere (4–6).

If brains are doing robust control using internal models, then illusions may be intrinsic. What reaches conscious awareness is the state of a simulation, not a direct perception of the world (9–12). However, seeing is believing, because what we see is not only a remarkably robust, integrated, dynamic state estimate of the external world blending multiple senses but one that automatically focuses attention on information that we need to take robust actions. That we can be almost arbitrarily fooled is one of the many unavoidable tradeoffs of our physiology, evolution, and brain architecture. Therefore, it is equally true that seeing is dreaming, which is known from a variety of well-studied illusions (3), dreams, and hallucinations. Functional losses because of CNS lesions (4, 6) are often highly specific and reproducible, making us aware of myriad unconscious processes that were previously taken for granted and showing that our internal simulations use a distributed and parallel implementation to mitigate the effects of hardware delays. The extreme gain and loss of capabilities in savants also suggest powerful but constrained simulation capabilities.

Computer and Control Technology

Modern robust control systems are typically implemented using digital hardware and software, and most computers are embedded in this way and thus, are permanently hidden. Examples are ubiquitous from antilock brakes to automated collision avoidance to global positioning systems in cars to fly by wire aircraft. Networks and cloud computing that connect the relatively fewer (but still billions worldwide) personal computers and smart phones have hidden routers and servers that control the flow of packets and files. The internal mechanisms are again manifest largely in the rarity of crashes, losses, errors, and failures and in the catastrophic nature of rare crashes. However, despite enormous progress, robots struggle to navigate the real world as effectively as rodents or even insects, and computers continue to fail in Turing tests, although in fascinating ways that reveal much about both humans and computers (42). This enormous, hidden, cryptic complexity, driven by robustness, is both the greatest initial obstacle in using advanced information and control technologies as metaphors for biology and also ultimately, the key to important insights and theories (16, 19).

As a starting point, human memory layering seems to be very different from computers, which is shown by various syndromes, lesions, and laboratory studies (6) as well as competitions pushing the extremes of human memory (14). A standard technique among competitors memorizing sequences of meaningless symbols is to embed them in previously prepared complex and vivid 3D dynamic simulations (called palaces) that can then be replayed to retrieve the symbols. For example, such methods allow experts to memorize a single pack of 52 shuffled playing cards with no errors in less than 22 s. Palaces are reused after the memories are actively purged. This finding illustrates that humans can repurpose innate (dynamic, modal, and grounded, etc.) simulation capability in lower layers for purely symbolic higher-layer memory and (it is claimed) the lack of real alternatives for rapidly storing and retrieving large amounts of symbolic data (14). That it works so poorly is perhaps less remarkable than that it works at all.

Computers have opposite memory capabilities from humans in that massive amounts of purely symbolic, meaningless data are nearly instantaneously found, stored, searched, and retrieved, and as a result, Google is now a verb. This finding is possible, because computers and networks have, arguably, the canonical

layered architecture in engineering from very-large-scale integration (VLSI) chip design to the transmission control protocol/internet protocol (TCP/IP) protocol stack (16, 19), and a brief look at such architecture is a rich source of insights. Near the bottom is analog circuitry that is exquisitely organized (extremely large/thin/nonconvex) to create digital behavior when interconnected appropriately but at the expense of speed and efficiency. These analog and digital hardware layers are functionally distinct but physically coincident. Importantly, these hidden layers and interfaces are fundamental to the more obvious plug and play modularity that they enable.

Many devices can be built purely out of hardware, but a software layer gives much greater flexibility at even more expense of speed and efficiency. As with the digital layer and its analog substrate, software only exists when embodied in hardware, but because software can be moved across hardware platforms, it also has an existence that transcends any individual physical instantiation. Software is a very special organization of hardware, and similarly, digital hardware of analog circuitry, but no simple terminology captures the full richness of this layering. Nevertheless, there are no mysteries here, just an impoverished language for description. Software, also, is richly layered. An operating system (OS) often has a kernel layer that manages and virtualizes the various hardware resources for higher-layer application programs. For example, the hardware memory typically has its own separate layering of memories from small, fast, and expensive to large, slow, and cheap. This layering is within the hardware, and therefore, it is orthogonal to that of analog to digital to software.

By managing the use of layered memory cleverly, the OS kernel can provide applications programs with a virtual memory that has nearly the speed of the fastest hardware, with the cost and size of the cheapest hardware. Such virtualization is a familiar and essential element of layering. Therefore, applications can use abstractly named variables and higher-level languages, and the kernel then translates these names into virtual addresses and ultimately, physical addresses; however, the name to address translation process is hidden from the applications. This OS architecture provides a variety of robustness features from scalability of the name and virtual address spaces to resource sharing between applications to security of the physical memory from application failures or attacks.

At the most basic level, the Internet TCP/IP protocol stack extends the functionality of the OS kernel across the network to multiple machines, allowing much broader resource sharing and creating the illusion to users of near infinite resources. Unfortunately, TCP/IP was designed decades ago not as a general purpose platform but primarily to be robust to physical attacks on hardware in relatively small networks with trusted users running minimal applications. It did this brilliantly, especially compared with the alternatives at the time, but modern use is largely the opposite. Hardware is more reliable than software, which is more trustworthy than users, and the network is large and supports a bewildering range of applications. In essence, TCP/IP is not strongly layered enough. It lacks a modern naming and virtual addressing mechanism, leading to problems with security, performance, scalability, multihoming, and mobility for which resolution is hotly debated even among experts (43). That TCP/IP is in some ways inadequate is less surprising than that it works at all given the astonishing change that it has enabled.

TCP/IP is an example of how architectures that are well-designed for extreme robustness can create evolvability as a side benefit, perhaps the essential benefit of good architectures and the focus of the rest of this paper. Networked and embedded control computers may ultimately be a good source of metaphor and theory for neuroscience, because we know exactly how the system behavior depends on the technical details, and a rich and growing body of mathematics formalizes the insights (19). Unfortunately, these details are, by design, largely hidden from users, and although experts will find the previous discussion trivial and obvious, many readers may not. Also, despite abundant relevant tutorial material on computer architecture (less on networks), there is little discussion on which of its features arose from fundamental design vs. historical accidents of rapid evolution. For these reasons, we explore some additional case studies that are transparent and familiar but illustrative of the fundamental concepts of complexity, architecture, layering, and robustness.

Layered Architectures Simplified

Clothing and textiles represent a simple case study in network architecture and the role of robustness and layering, with paper as a special case. Although clothing may seem a frivolous illustrative example, it is based on many levels of complex technologies and reveals universal organizational principles, with details that are easily accessible to nonexperts. On the surface, each clothing module or garment (coat or socks) looks fairly similar, hiding chemical and physical differences in weave, elasticity, water-resistance, breathability, UV protection, and even insect repulsion. The constraints imposed by fashion trends on the success of clothing as a technology illustrate important points but will be deemphasized, and our consideration of the architecture of an outfit focuses more on essential function and robustness in harsh environments. The basic function of clothing is protection, providing comfort over a wide variety of external (weather and temperature) and internal perturbations (physical activity). Like other complex systems, complexity in clothing is driven by robustness to extremes more than by need to provide minimal function. Human skin seems optimized by evolution for dissipating heat during endurance running in the tropics (44–46), and it offers little protection compared with heavy fur. Clothing provides that protection when needed.

Four fairly universal layers exist within textile architecture: (*i*) fibers that are spun into (*ii*) yarn or thread, which are woven or knitted into (*iii*) cloth that is sewn into (*iv*) garments. Cotton fibers are about 12–20 μm in width and several centimeters in length, roughly comparable with large neurons but with very different morphologies; 1 kg cotton has less than 1 billion fibers, large but still much less than the number of neurons in 1 kg brain, and the way in which fibers are interconnected is much simpler than neurons. Thus, in this simple but easily understood example of layering, the properties of textiles are not obvious from those properties of fibers. Tens to hundreds of fibers are spun into yarn and thread of essentially arbitrary length, which are woven or knitted into cloth that is nearly 2D and sewn into garments, also of arbitrary size.

This layered construction is much simpler, but it parallels analog to digital hardware to software and the polymerization of metabolic building blocks to macromolecules that assemble into networks and cells. In all of these examples, the layered architecture illustrates universal principles of organization and protocols for construction. Each layer has the large/thin property for which functional alternatives are almost unaccountably numerous but are nevertheless a vanishingly small fraction of all possible (e.g., random) configurations. Each layer must be exquisitely organized to produce the layer above it, which is not necessarily physically distinct. Garments are functionally distinct from the fibers from which they are physically composed, which is the same for cloth and yarn.

The complexity of the textile architecture is driven by robustness tradeoffs, because all of the layers from fiber to cloth can be completely collapsed to make paper, a nearly random connection of fibers with no intermediate layers. Paper is an extreme example of a degenerate special case of a layered architecture. Here, degenerate simply means that the constraints that define the architecture are relaxed or removed entirely.

Additionally, with minimal additional complexity, paper can be sewn into specialized but unavoidably fragile garments. This finding makes clear that the complex internal, hidden layering is only for robustness and is not needed for minimal functionality, because paper can easily stand in for cloth in idealized environments. Similarly, small bio-inspired networks of metabolites and enzymes can be used to manufacture valuable chemicals, but they lack the robustness and evolvability of whole organisms. The overall textile architecture has persisted for many thousands of years (the bacterial cell for billions of years), whereas technologies within layers evolved rapidly.

One feature of the fiber to garment (i.e., garment/sew/cloth/weave/thread/spin/fiber) layered architecture is that it is robust enough that we can temporarily defer its study and focus on something even simpler, which is how individual garments are also layered to make outfits. Outfits for harsh environments typically have three kinds of layers. The outer shell layer protects from wind and water, the middle or insulation layer provides warmth, and the inner or base layer is comfortable next to the skin and keeps it dry. These three layers each are composed of garments, which have within them the fiber to garment layers. Therefore, the garment to outfit layering and the fiber to garment layering are, in some sense, orthogonal, although there is no standard terminology. This finding illustrates an almost trivial but nevertheless crucial feature of organized complexity. Because this overall architecture is intrinsically so robust, we can temporarily take 1D (fiber to cloth layering) for granted and view it as a platform for another simpler dimension of clothing, but one that also connects with more popular views of modularity, while still introducing some essential elements of architecture.

Layering of garments to make outfits is one obvious architectural feature of clothing providing robustness to environments. These modular layers are physically distinct (unlike fiber to cloth) and can be shed or reincorporated as needed. This finding has obvious parallels with software. Good programming practice includes breaking large algorithms into smaller subroutines with simple interfaces; this modularity is more familiar but less fundamental than the layering of analog to digital hardware to software that makes it possible in the first place. In the absence of robustness requirements, the necessary engineering aspects of architecture can recede, and clothing can become considerably simplified or elaborate (as dictated by fashion). In perfect environments, little or no clothing is required.

Similarly, even the most complex architectures allow for much simpler degenerate special cases (analogous to paper) under idealized circumstances. Tradeoffs abound within each layer in fabric, weight, cost, durability, and fasteners, etc. With changes in technology or conditions, garments can become obsolete or evolve (e.g., body armor and Velcro, etc.). Highly optimized, robust, and efficient garments that are finely specialized for a specific layer, body part, and individual are typically fragile to other uses (other layers, body positions, or wearers of different size and shape) and may be costly. Simple wraps or rags are very versatile but at best, yield outfits that are fragile to environment and movement. Most garments fall between these extremes.

If we knew nothing about the layering of garments, we might learn little from observing intact outfits, but we could begin reverse-engineering the architecture through lesions or knockouts in controlled experiments. These experiments might require harsh experimental conditions and perhaps, appropriately instrumented crash dummies. Damage or loss of a garment layer can cause very specific loss of robustness: outer layer to wind and/or water, middle layer to cold, and inner layer to comfort. Changes to fiber types, yarn, or sewing could be lethal at different levels, revealing their functional role. Most informative would be small changes with large consequences, such as unraveling a seam to reveal the role of sewing in garment construction or disruption of a weave or knit to reveal its role in cloth integrity.

Architecture as Constraints That Deconstrain

The view of architecture as constraints that deconstrain (17, 18) originated in biology, but it is consistent with engineering (16) and illustrated by clothing. A robust architecture is constrained by protocols, but the resulting plug and play modularity that these shared constraints enable deconstrain (i.e., make flexible) systems designed using this architecture. Constraints give a convenient starting language to formalize and quantify architecture and ultimately, a mathematical foundation (19). Concretely, consider a given wardrobe that is a collection of garments and the problem of assembling an outfit that provides suitable robustness to the wearer's environment. Three distinct but interrelated types of constraints are universal in clothing as in all architecture (16): (*i*) component (garment) constraints, (*ii*) system (outfit) constraints, and (*iii*) protocol constraints. Therefore, in combination, diverse, heterogeneous components (garments) that are constrained by materials and construction combine synergistically (through protocols) to yield outfits that satisfy system constraints not directly provided by any single component. We will use outfit to describe a functional, robust set of garments, and heap (Craver uses aggregates) (47) to describe a random collection not required to have any other system features.

The protocols that constrain how garments make outfits are simple and familiar, and a minimal view is that each of g garment categories (e.g., socks, sweaters, coats, boots, and hats) is constrained to a specific layer and body position and thus, to a specific and essentially unique location within an outfit. Suppose, for simplicity, that a wardrobe has n garments of each type for a total of ng garments. If any of the n garments of a specific type can be part of an outfit, then there are a total of $F = n^g$ possible outfits. For example, for $n = g = 10$, there are $ng = 100$ total garments but 10^{10} (10 billion) distinct outfits that obey the layered architecture. However, there are 2^{ng} subsets or heaps of ng garments, and therefore, if protocols are ignored, the number of unconstrained garment heaps is vastly larger. For $n = g = 10$, there are $2^{ng} > 10^{30}$ such heaps, and therefore, heaps chosen without regard to protocols have a vanishingly small chance of being outfits (another example of large/thin).

The discussion of clothing so far provides only a static view of architecture. In reality, the core of good architecture is ability to facilitate change over many timescales, including overall architecture (millennia), manufacturing technology (centuries), garments (decades), and outfits (daily). This example can be expanded to hint at the dynamic and control dimensions of both us and our clothing. Well-constructed outfits respond so automatically to movement that wearers can normally ignore the hidden internal complexity that makes this possible, just as we do the control of movement itself.

The roughly minute to hour timescales needed to assemble outfits also illustrate the role of dynamic control within architecture. The most obvious control is the actual forward assembly of a specific choice of g garments into a layered outfit. Most of the protocols that govern this process are readily learned by children, although specialized garments may require complex control (e.g., bowties and shoe laces). Ideally, the protocols are complete enough that any outfit made that obeys them will automatically satisfy system constraints (this is rare in engineering, because it is hard to design protocols with such guarantees). Humans easily visualize (simulate) what an outfit will look like from seeing the separate garments, but often, they still need to try them on to be sure of details of fit and appearance. Our simulators are robust but imperfect.

More subtle and complex (and less easily learned) is the backward process of choosing these garments to match the day's specific systems constraints, which are most dependent on weather and the wearer's activities. This process takes simulation to another level. Because layering allows dynamic reconfiguring

of an outfit in real time, this backward selection control or management process potentially interacts with forward assembly control on all time scales. The backward process of choosing components is typically much more complex (and less obvious) than the forward assembly process, and this process can lead to confusion about the role of control and feedback in architectural design. Hopefully, in this concrete example, the processes are obvious, even if the language for describing it is inadequate.

On longer time scales (days to decades), the user might assemble a wardrobe of garments, again guided by the overall architecture by which garments make outfits. Manufacturing technologies can change on year to century timescales but must both reflect the architecture and only slowly change aspects of it. New technologies (such as spandex and Velcro) relax component constraints and allow new systems without fundamentally changing the architecture of clothing, which has persisted for millennia. Here again, in the engineering context, robustness largely drives complexity, because without changing system and component constraints, the protocols and control processes for connecting and reconciling them could be vastly simpler (e.g., standardized uniforms).

Bowties, Hourglasses, Pathways, Flows, and Control

Another aspect of constraints that deconstrain is the relatively small diversity in the protocols and processes that connect layers and enable vastly greater diversity in the materials that constitute a layer. This aspect of architecture can be visualized as a bowtie or hourglass (depending on whether layers are visualized horizontally or vertically) (27). For example, the fairly universal, homogeneous process of sewing (the bowtie knot or hourglass waist) takes an almost unaccountably greater and extremely heterogeneous diversity of cloth (fanning in to the knot) into an even greater diversity of garments (fanning out of the bowtie). Similarly, the reactions and metabolites of core metabolism are largely universal, connecting extremely diverse layers of catabolism to moderately diverse biosynthesis. The basic processes and codes underlying transcription and translation are highly conserved, but the specific genes and gene products are extremely diverse.

After the diverse garments sewn from cloth become components in the layered outfit architecture, they are categorized into the much less diverse types of garments and assembly protocols that, in turn, make a hugely diverse set of outfits. Threads and yarns are more diverse than the few canonical weaves, knits, or knots that create diverse textiles. The least diversity is in the fibers (from plant, animal, mineral, and synthetic origins) and the spinning processes that make yarn and thread. (What are vastly diverse are the geographic origins of these fibers.) Thus, the great diversity and heterogeneity within each layer also varies among layers, and it even depends on the categories used to define diversity. Garments (or cells) are unaccountably diverse and deconstrained when viewed in detail as the result of the garment/cloth/yarn/fiber (or DNA/RNA/protein) architecture but much less so when viewed as satisfying system constraints of that architecture, the component constraints of the outfit architecture, or the constraints on cells or cell types. The protocols between layers are typically more fixed and much less diverse by any measure than the layers that they connect.

Because they are the fixed points in robust architecture, when protocols are subject to attack, the system can fail catastrophically. Seaming is the protocol that sews fabric into garments, providing structure and function. Seams make the important difference between wraps and clothes, but they are the main source of clothing's fragile robustness. If the seam connecting the shoulder to sleeve unravels, a coat is useless. The greatest fragility of universal knots/waists is that they facilitate hijacking and attack by parasites and predators. Viruses hijack cellular transcription/translation machinery, and predators exploit the fact that they share universal and essential metabolic building blocks with their prey. In neuroscience, the role of dopamine in a robust and flexible reward system (knot of the bowtie) is fragile to hijacking by addiction (5). The stitches used in seams are not as tight as fabric weave and therefore, seams are internalized and often protected by lining, illustrating how good architectures allow hiding of necessary fragilities as much as possible. Our skulls, cardiovascular system, blood–brain barrier, and immune systems similarly protect fragilities of our brains to trauma, intense activity, and infection at the expense of the overhead to maintain them.

In both textiles and biology, obvious natural pathways and flows of materials and information assemble systems from components. Indeed, depicting these architectures in terms of pathways rather than layers has been the dominant view in science (and until recently, in engineering as well). Although not inconsistent with layering, the emphasis on a pathway view has limited our understanding of control, complexity, and robustness. Although the interplay between computational complexity, constrained optimization, and robust control has been deeply explored in the last decade, with broad applications including the Internet (19), power grids, and systems biology, no universal and accessible taxonomy for describing these various flows and their various complexities has emerged, even within engineering.

We already saw that the backward process of deciding on an outfit that satisfies the constraints of a given activity and environment is vastly more complex than the forward process that assembles the outfit from a set of garments. Similarly, the feedback control (e.g., of looms and sewing machines) within each layer of textile manufacturing is vastly more complex than the forward flow of materials from fibers to textiles. Additional complexity comes from the backward flow of textile design that turns constraints on textiles into specifications on the manufacturing processes as well as the supply chain management of the resulting process in response to customer demand.

Biology has similarly complex feedbacks. There are 10 times as many fibers feeding back from the primary visual cortex to the visual thalamus as there are in the forward flow (6). Wiring diagrams that include both autocatalytic (e.g., of ATP, NADH, etc.) and control feedback in metabolism are so much more complex than the usual depictions of relatively simple tree-like flows of metabolites that they are rarely drawn in detail except at the level of small circuits. Even complete wiring diagrams do not reflect the true complexity of control (20). The control of transcription and translation is vastly more complex than the basic forward polymerization processes themselves. Although we have no difficulty understanding the basic nature of these feedbacks and their roles in these specific architectures, the lack of an adequate language to generalize and/or formalize is still a roadblock, especially because engineering jargon is domain-specific and heavily mathematical.

Hidden Complexity, Illusions, and Errors

Related but important research themes can be mentioned only briefly while recapping the main points. For example, the emphasis on dynamic and mechanistic explanations in the philosophy of neuroscience (47–49) is compatible, is complementary, and hopefully, can help lead to a more coherent and consistent shared language, which is desperately needed. The dangerous illusions and errors that plague individuals are often amplified by institutions, and this finding is relevant to engineering as well, because policy and politics often trump technology (39). Arguably, the most dangerous and pervasive of popular illusions is that our actions are unconstrained by hard tradeoffs, a problem increasingly acute in everything from teaching evolution to dealing with global warming. A unique case study in human error, because it is entirely within science, is the genre of research that has dominated mainstream literature for decades under the

rubric of new sciences of complexity and networks (NSCN). NSCN is relatively new in neuroscience, but it already has an appealing narrative (50) and extensive and accessible reviews (51, 52). Claims that NSCN has been a success in other fields (50–52) are supported by impact factor measurements, but NSCN has led to persistent and systematic errors and confusion that show no sign of abating (refs. 16 and 30–32 and references therein). Although the goals of NSCN research are somewhat consistent with our goals, particularly in neuroscience, the methodology is not. Most concepts and terminology in NSCN do not overlap with control theory, but those terms that do overlap can have opposite meaning. A thorough discussion is beyond the scope of this paper, but a simple instance is illustrative.

Engineered and biological systems necessarily make ubiquitous use of nonlinearity, recursion, feedback, and dynamics, which in NSCN, are almost synonymous with unpredictability, fractals, self-similarity, or chaos (16). In engineering, quite the opposite is true. For example, the amplifiers in the sensors and actuators that enable robust controllers are necessarily extremely nonlinear. Digital computers are recursive and also extremely nonlinear (i.e., switching using transistor amplifiers in hard saturation) but are the most repeatable complex systems that we build. Ironically, to a naïve observer, the analog behavior of the billions of transistors (each much smaller and simpler than a neuron) per chip in a computer might seem bewilderingly noisy, chaotic, and unpredictable, with no hint of the almost perfectly robust, repeatable digital behavior that results at the interfaces and that engineers use to build systems. Of course, this finding is exactly the purpose of the very special large/thin/nonconvex organizations that constitute digital/analog and other forms of layering. This finely tuned, hidden diversity and complexity underlying robust systems are also the opposite of complexity in NSCN, which emphasizes minimally tuned, mostly random interactions of large numbers of homogeneous components that yield surprising emergent self-organization and order for free (50–52).

Recall that, in the clothing example, for a fixed number of garment types g, the number of outfits $F = n^g$ is constrained to polynomial growth in the number n of each garment type, whereas all possible heaps grow exponentially in n. A related source of confusion in biology is whether biology is fine-tuned vs. robust, as if these types were mutually exclusive. There is robustness in the large (polynomial growth) sets of structured and functional networks and fine-tuning that makes these sets a thinly small subset of the vastly larger (exponential growth) set of random nonfunctional networks. Highly evolved biological systems are large/thin and both fine-tuned (obey strict and far from random protocol constraints) and robust. Indeed, this finding is the essence of constraints that deconstrain and a necessity, not a paradox. The connection between the large/thin feature of neural circuits (1) and robust control is even deeper, which is sketched above.

In a completely different direction, research on human evolution has recently exploded in both depth and accessible exposition (44–46), and the picture emerging is also complementary and compatible to ours. Compared with other great apes and top predators, humans are physically weak and slow with thin skin, no protective fur, and small guts that digest raw food poorly, all possibly fragile side effects of evolved robustness to running long distances in hot weather (46). When paired with even minimal technologies of weapons (e.g., simple sticks and stones), fire (for cooking, protection, and warmth), and teamwork, we go from helpless prey to invincible top predators (45), whose main threat then comes from other, similarly equipped humans. Our layered biological architecture of brain to mind is now augmented by layered technological architectures such as fiber to garment to outfit, metal to tool, weapon, machine, analog to digital, hardware to software, all of which expand the cognitive niches (44) in which we can robustly function. We have now eliminated our fragilities to our environment but replaced them with new and potentially catastrophic fragilities of our own making.

The tradeoffs that we see throughout these architectures and systems between efficiency and robustness, as well as between robustness to various different perturbations, are necessities and not accidents, although choices are still abundant within the resulting constraints. Versions of such tradeoffs can be formalized and made mathematically precise (20), and *SI Text* presents a simplified tutorial on the mathematics and model systems. More speculative but plausible is the claim that layered architectures are also necessary to effectively balance these tradeoffs, which is evidenced by their ubiquity in biology, physiology, and technology. That is, inner/lower layers are large/thin/nonconvex and must remain hidden within the system for robustness. For example, inner and middle garments must remain hidden behind the outer shell to provide comfort and warmth in a harsh environment, whereas the middle and outer layers must be segregated from skin for comfort. Because each garment maintains a separate identity that can be easily recovered by disassembling an outfit (or heap), this finding is perhaps the paradigm for modularity (51), but it is a very special case and on its own, quite misleading without considering the protocol and component constraints.

The component constraints on garments within the shell/insulation/base layering of outfits depend on material properties that derive from the orthogonal garment/cloth/yarn/fiber layering of textiles. In these orthogonal layers, garments are very special large/thin/nonconvex organizations of fibers/threads that must lose their individual identity. Loose threads (disobeying protocols even minimally) make garments fragile. Similarly, cells are a special organization of macromolecules, digital hardware of analog circuitry, software of hardware, brains of cells, and perhaps, minds of brains. If the brain is layered at the highest modular level (6, 15), roughly analogous to outfits/garments, then there is an orthogonal layering of tissues down to cells down to macromolecules that occurs within each of the macro brain layers, analogous to the garment to fiber layering.

In all of these layered architectures, the virtualization of the lower layer resources is an illusion that can be maintained almost perfectly in normal operations of minds, outfits, machines, and computers. The hidden complexity is primarily needed to create this remarkable robustness and evolvability, not minimal function, and it is only revealed by pushing systems to their extremes by perturbing the environment or components in lower layers outside the constraints that the systems evolved to handle (3–6). After the layered architectures are in place, both our minds and software are free to rapidly evolve independently given the right environment, although the large differences between digital and brain hardware imply very different constraints (6). This plasticity is one of the main benefits deconstrained by the constraints of the bowtie/hourglass protocols that create the layering (17, 18, 27). The mind/brain layering is much more complex than the most complex current technology of embedded and/or networked software/hardware/digital/analog, but the latter would be utterly incomprehensible without the right conceptual framework, mathematics, and tools, most of which are still unknown or relatively new to neuroscience (7). We hope that the connections between neural and technological architectures will help to demystify some aspects of this complex continuum.

ACKNOWLEDGMENTS. We thank Mike Gazzaniga and Scott Grafton for helpful discussions and feedback. This work was partially supported by the National Science Foundation, the National Institutes of Health, the Air Force Office of Scientific Research, and the Institute of Collaborative Biotechnologies (Army Research Office W911NF-09-D-0001).

1. Marder E (2011) Variability, compensation and modulation in neurons and circuits. *Proc Natl Acad Sci USA*, 10.1073/pnas.1010674108.
2. Purves D, Wojtach WT, Lotto RB (2011) Understanding vision in wholly empirical terms. *Proc Natl Acad Sci USA*, 10.1073/pnas.1012178108.
3. Chabris C, Simons D (2010) *The Invisible Gorilla: And Other Ways Our Intuitions Deceive Us* (Crown Publishing Group, New York).
4. Gazzaniga M (2011) *Who Is in Charge: Free Will and the Science of the Brain* (Harper Collins, New York).
5. Linden DJ (2011) *The Compass of Pleasure: How Our Brains Make Fatty Foods, Orgasm, Exercise, Marijuana, Generosity, Vodka, Learning, and Gambling Feel So Good* (Viking Press, New York).
6. Eagleman D (2011) *Incognito: The Secret Lives of the Brain* (Pantheon Books, New York).
7. Nagengast AJ, Braun DA, Wolpert DM (2010) Risk-sensitive optimal feedback control accounts for sensorimotor behavior under uncertainty. *PLoS Comput Biol* 6:e1000857.
8. Grafton ST (2010) The cognitive neuroscience of prehension: Recent developments. *Exp Brain Res* 204:475–491.
9. Cisek P, Kalaska JF (2010) Neural mechanisms for interacting with a world full of action choices. *Annu Rev Neurosci* 33:269–298.
10. Körding K (2007) Decision theory: What "should" the nervous system do? *Science* 318: 606–610.
11. Shadmehr R, Smith MA, Krakauer JW (2010) Error correction, sensory prediction, and adaptation in motor control. *Annu Rev Neurosci* 33:89–108.
12. Barsalou LW (2008) Grounded cognition. *Annu Rev Psychol* 59:617–645.
13. Pinker S (2007) *The Stuff of Thought* (Penguin Group Inc., New York).
14. Foer J (2011) *Moonwalking With Einstein: The Art and Science of Remembering Everything* (Penguin Press, New York).
15. Hawkins J, Blakeslee S (2004) *On Intelligence: How a New Understanding of the Brain Will Lead to the Creation of Truly Intelligent Machines* (Times Books, New York).
16. Alderson DL, Doyle JC (2010) Contrasting views of complexity and their implications for network-centric infrastructures. *IEEE Trans Syst Man Cybern A Syst Hum* 40: 839–852.
17. Kirschner M, Gerhart J (1998) Evolvability. *Proc Natl Acad Sci USA* 95:8420–8427.
18. Kirschner M, Gerhart J (2005) *The Plausibility of Life* (Yale University Press, New Haven, CT).
19. Chiang M, Low SH, Calderbank AR, Doyle JC (2007) Layering as optimization decomposition: A mathematical theory of architecture. *Proc IEEE* 95:52–56.
20. Chandra F, Buzi G, Doyle JC (2011) Glycolytic oscillations and limits on robust efficiency. *Science* 333:187–192.
21. Doyle JC (1978) Guaranteed margins for LQG regulators. *IEEE Trans Automat Contr* 23:756–757.
22. Glover K, Doyle JC (1988) State-space formulas for all stabilizing controllers that satisfy an H-infinity-norm bound and relations to risk sensitivity. *Syst Control Lett* 11: 167–172.
23. Doyle JC, Francis BA, Tannenbaum A (1992) *Feedback Control Theory* (Macmillan, New York).
24. Zhou K, Doyle JC, Glover K (1996) *Robust and Optimal Control* (Prentice Hall, Englewood Cliffs, NJ).
25. Yi TM, Huang Y, Simon MI, Doyle J (2000) Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proc Natl Acad Sci USA* 97:4649–4653.
26. Csete ME, Doyle JC (2002) Reverse engineering of biological complexity. *Science* 295: 1664–1669.
27. Csete ME, Doyle JC (2004) Bow ties, metabolism and disease. *Trends Biotechnol* 22:446–450.
28. Doyle JC, Csete ME (2005) Motifs, control, and stability. *PLoS Biol* 3:e392.
29. Doyle JC, et al. (2005) The "robust yet fragile" nature of the Internet. *Proc Natl Acad Sci USA* 102:14497–14502.
30. Keller EF (2005) Revisiting "scale-free" networks. *Bioessays* 27:1060–1068.
31. Lima-Mendez G, van Helden J (2009) The powerful law of the power law and other myths in network biology. *Mol Biosyst* 5:1482–1493.
32. Willinger W, Alderson D, Doyle JC (2009) Mathematics and the internet: A source of enormous confusion and great potential. *Not Am Math Soc* 56:586–599.
33. Schulz K (2010) *Being Wrong: Adventures in the Margin of Error* (Ecco Press, Hopewell, NJ).
34. Heffernan M (2011) *Willful Blindness: Why We Ignore the Obvious at Our Peril* (Walker Books, New York).
35. Freedman DH (2010) *Wrong: Why Experts Keep Failing Us—and How to Know When Not to Trust Them* (Little, Brown, Boston).
36. Ioannidis JPA (2005) Why most published research findings are false. *PLoS Med* 2:e124.
37. Trikalinos NA, Evangelou E, Ioannidis JP (2008) Falsified papers in high-impact journals were slow to retract and indistinguishable from nonfraudulent papers. *J Clin Epidemiol* 61:464–470.
38. Stiglitz J (2010) *Freefall: America, Free Markets, and the Sinking of the World Economy* (W. W. Norton & Company, Inc., New York).
39. Oreskes N, Conway EM (2010) *Merchants of Doubt: How a Handful of Scientists Obscured the Truth on Issues from Tobacco Smoke to Global Warming* (Bloomsbury Publishing, London).
40. Buffett HG (2009) *Fragile: The Human Condition* (National Geographic Books, Des Moines, IA).
41. Kelly K (2010) *What Technology Wants* (Penguin Group, Inc., New York).
42. Christian B (2011) *The Most Human Human: What Talking with Computers Teaches Us About What It Means to Be Alive* (Doubleday, New York).
43. Day J (2008) *Patterns in Network Architecture: A Return to Fundamentals* (Prentice Hall, Englewood Cliffs, NJ).
44. Pinker S (2010) Colloquium paper: The cognitive niche: Coevolution of intelligence, sociality, and language. *Proc Natl Acad Sci USA* 107(Suppl 2):8993–8999.
45. Wrangham R (2009) *Catching Fire: How Cooking Made Us Human* (Basic Books, New York).
46. Bramble DM, Lieberman DE (2004) Endurance running and the evolution of Homo. *Nature* 432:345–352.
47. Craver CF (2007) *Explaining the Brain: Mechanisms and the Mosaic Unity of Neuroscience* (Oxford University Press, New York).
48. Waskan JA (2006) *Models and Cognition: Prediction and Explanation in Everyday Life and in Science* (MIT Press, Cambridge, MA).
49. Kaplan DM, Bechtel W (2011) Dynamical models: An alternative or complement to mechanistic explanations? *Top Cogn Sci* 3:438–444.
50. Chialvo DR (2010) Emergent complex neural dynamics. *Nat Phys* 6:744–750.
51. Bassett DS, Gazzaniga MS (2011) Understanding complexity in the human brain. *Trends Cogn Sci* 15:200–209.
52. Bullmore E, et al. (2009) Generic aspects of complexity in brain imaging data and other biological systems. *Neuroimage* 47:1125–1134.

Supporting Information

Doyle and Csete 10.1073/pnas.1103557108

SI Text

This supporting information has some very elementary tutorial introductions to minimal control theory, with details that are easily checked and experiments that are easily performed. The larger research program that we are pursuing can be summarized as if you accept the empirical and theoretical claims in neuroscience, then what is the right mathematical framework to connect the mechanistic details with the behavior? Tremendous theoretical progress is being made on what we believe is the right combination of robust, layered, distributed, and nonlinear control, integrated as needed with communication and computation theories. At best, the state of the art is nascent and promising but technically inaccessible, and it has been focused on computer and energy networks, not neuroscience. Because the core theory is being developed and vetted among control theorists, we will aim to build language (not standardized or well-thought out so far) that can be used to connect the theory with real behavior and known constraints on the components. Essential to this dialogue is a shared understanding of layered architectures, robustness, feedback, dynamics, and optimization as well as why a constraints-based theory is both theoretically and biologically natural. What follows are simple examples that complement the layering ideas in the text.

Minimal Control Theory

The amplifier circuit in Fig. S1 is the simplest feedback system possible, and it will be used as a minimal starting point to start formalizing aspects of the constraints view of architecture motivated by the case studies in the paper. Then, we will briefly sketch out the theory underlying the pendulum case study, which is much more technical but still uses only undergraduate mathematics.

There are amplifiers loosely based on the principle described below all around us. Most familiar might be the audio amplifiers and tuning knobs for radios and audio equipment. The large/thin element is that, say, a high-tech radio has many knobs to adjust to get the station, volume, compensation for room acoustics, etc. Thus, there is a nearly infinite range of functional parameter values for the radio that can be explicitly tuned with just a few knobs. However, if you break open the box, there is a much larger number of hidden parameters whose variation can destroy the function inside. These variations are not allowed in normal use, and this use would be breaking the system. Even minor but random rewiring of the internal connections will ruin the radio. Also, if you make a math model of the circuit, the internal parameters are infinitely more numerous than the external tunable ones, and the math model will have the property that the set of functional parameters is both vastly large in absolute terms and also vanishingly thin as a fraction of all possible (random) perturbations.

Fig. S1 is the simplest possible math model that illustrates roughly how real feedback amplifiers work. The signals are an external input r and output z, an amplifier noise n, internal measurement y, and control u. The components are an (open-loop) amplifier A and controller C, giving the interconnected system output/input z/r gain G and noise response S. All of these quantities are assumed to be (possibly uncertain) real numbers, which capture the small signal (linear) steady state features of the circuit. It is the interplay between feedback and dynamics that is, ultimately, of most interest but also most confusing; therefore, we will first explore feedback without dynamics as a small step.

The basic properties of this circuit can all be derived using elementary algebra, including solving for the relationship among the external signals $z = Gr + Sn$ as (Eq. S1)

$$\begin{aligned} &z = A(r+u)+n = A(r+Cz)+n \\ &\Rightarrow z = \left(\frac{1}{1-AC}\right)(Ar+n) \triangleq Gr+Sn. \end{aligned} \qquad \text{[S1]}$$

Powerful but precise amplifiers seem to be essential throughout technology and biology and are, in any case, ubiquitous. What is typically desired in electrical, mechanical, and/or hydraulic technologies as well as in control muscles (A) with nerves (C) is that the gain G be large and precise and the noise response S small. This model is an extremely abstracted and minimal model of such systems.

If there are no constraints and no uncertainty (i.e., A and C can be specified arbitrarily and exactly and the noise $n = 0$), then feedback is unnecessary, and any gain G can be realized perfectly with amplifier $A = G$ and $C = 0$. Real high-gain amplifiers unavoidably have both noise and limits on their gains, and therefore, feedback is necessary; there are also limits on the achievable closed-loop performance and robustness that we will explore. There are, again, three distinct but interrelated types of constraints: (i) component constraints, (ii) system constraint, and (iii) protocol constraints. Mathematically, a minimal set of components constraints would be (S2)

$$\text{Component: } A \geq A_{\min},\ 0 < -C < 1,\ n \neq 0. \qquad \text{[S2]}$$

Typically, the noise n would have more characterization, but for our purposes here, its presence or absence is all that will be considered. Also, $A \geq A_{\min} >> 1$ would be a high but uncertain amplifier gain, with only the lower-bound $A_{\min}$ assumed to be known (and thus, an upper bound on A is not assumed), whereas C is a (negative) feedback that has small gain, which can be tuned exactly to any desired value $0 < -C < 1$. In practice, C might be implemented with a potentiometer to give a (precisely) variable gain amplifier, and this finding is roughly what a volume knob on a radio does. These are typical tradeoffs that appear in real elementary components that can have either high or precise gains but not both. In all cases, larger $A_{\min}$ would be more costly (e.g., larger and/or require a larger power supply), and therefore, the choice of $A_{\min}$ would be a key design decision involving tradeoffs between these factors. The amplifier is also assumed to have noise $n \neq 0$, which will force constraints on S.

The corresponding minimal constraints on the system can be expressed as (Eq. S3)

$$\text{System: } S \leq S_{\max}, G \in [G_{\min}, G_{\max}] = G_{\max}[(1-S_{\max}), 1]. \qquad \text{[S3]}$$

What is typically required is to have moderately large and bounded gain $G \in [G_{\min}, G_{\max}]$, with small gain $S \leq S_{\max}$ on the noise n. Note that, in this simplified formulation, S (called the sensitivity function in control theory) measures both the effects of noise on the output and the uncertainty in the gain $G \in [G_{\min}, G_{\max}] = G_{\max}[(1 - S_{\max}), 1]$, whereas in general, these numbers could be different. Robust systems would have $S_{\max} << 1$, which rejects noise and has small gain error, and high-performance systems would also have $G_{\min} >> 1$. The protocol constraints, which are described algebraically as (Eq. S4)

$$\text{Protocol: } G = AS,\ S = \frac{1}{1 - AC}, \qquad \textbf{[S4]}$$

are depicted schematically in Fig. S1 (deliberately drawn to emphasize that lower and higher are purely conventional and have no intrinsic meaning).

If we start with the component and protocol constraints in expression **S2** and Eq. **S4**, we can easily derive the resulting system constraints as (Eq. **S5**)

$$S \le S_{\max} = \frac{1}{1 - A_{\min} C},\ G \in [G_{\min}, G_{\max}] = \frac{1}{-C}[(1 - S_{\max}), 1]. \qquad \textbf{[S5]}$$

Engineers call this process analysis, because it takes a given set of component constraints and their interconnections (from the protocols) and analyzes the resulting system properties, which are also naturally expressed as constraints. It is the obvious forward flow from components through protocols to systems described above for textiles and biology. In this context, the constraints in Eq. **S5** are the consequences of those constraints in expression **S2** and Eq. **S4**. However, we can also go the other way (backwards), starting with the system constraints in Eq. **S3** and using the protocol constraints in Eq. **S4** to synthesize or design the component constraints by solving for A and C to get (Eq. **S6**)

$$S \le S_{\max} < 1, G \in [G_{\min}, G_{\max}],\ 1 < G_{\max}, G_{\min} = (1 - S_{\max}) G_{\max} \Rightarrow C = \frac{-1}{G_{\max}} \Rightarrow A_{\min} \ge \frac{-1}{C}\left(\frac{1}{S_{\max}} - 1\right). \qquad \textbf{[S6]}$$

Engineers call this synthesis, because it involves starting with system requirements and architecture (i.e., protocol constraints) and synthesizing the necessary component constraints. Analysis can then be used to verify that this synthesis was successful. In this case, both directions are trivial, but synthesis is typically vastly harder and more complex than analysis, an issue not well-illustrated by this simplified example. Recall that this example is similar to the case studies in the paper where the feedbacks were far more complex than the more obvious forward assembly pathway. The iterative process of synthesis (usually using greatly simplified models and constraints) and analysis (with more complete models) contributes to system design, which in realistic situations, can involve many additional processes not considered here.

Where causality arises in Fig. S1 is in certain relationship between signals and systems. For example, the control input u can reasonably be said to be caused by the combination of controller C and signal y through $u = Cy$. A more subtle point that we are glossing over here is that the signals and systems in Fig. S1 are abstract objects, and they correspond to a functional decomposition of the system, not a physical one. That is, a physical circuit implementing Fig. S1 would not break up into the physical modules of controller and amplifier any more than a digital circuit is physically distinct from its analog implementation. Fortunately, this is a subject with abundant tutorial material, and therefore, the interested reader can easily find accessible explanations.

Beyond this trivial sense, the scientific jargon of causation and emergence provides little here, whereas the processes of analysis and synthesis are clearly defined after the three types of constraints are clarified. Neither controller nor amplifier causes the system behavior, and the protocols in this particular architecture imply a logical connection between component and systems constraints that can be used in either direction (i.e., analysis vs. synthesis). If we ask why there are the specific component constraints in expression **S2**, a proximal answer is that the underlying technology makes it possible to fabricate components with these constraints and features. A more complete answer is that this technology also allows the systems constraints in Eq. **S5** to be realized through the protocols in Eq. **S4**. If these constraints are not compatible, the architecture as specified is not viable, and much of engineering involves evaluating this possibility. Although it is possible to describe this idea in terms of up and down causation or emergence, these terms seem to add nothing to our understanding.

With these preliminaries, we can now briefly discuss design tradeoffs and parameter spaces. Fig. S2 plots the tradeoffs between system performance in terms of $G_{\min}$, $G_{\max}$, and $S_{\max}$ as a function of C for $A_{\min} = 100$. In Fig. S2 *Right*, note that, for positive feedback $C > 0$, $S_{\max} > 1$ and noise is amplified. Thus, $C < 0$ is necessary and sufficient for $S_{\max} < 1$, a minimal robustness requirement. Given $C < 0$, then $\frac{1}{A_{\min}} \ll |C|$ (equivalently $|CA_{\min}| >> 1$) is necessary and sufficient for $S_{\max} << 1$, in which case $S_{\max} = |S_{\max}| \approx |\frac{1}{CA_{\min}}| \ll 1$. However, for a given $A_{\min}$, there is a tradeoff between making $|S_{\max}|$ small and $G_{\min}$ large. Robust and functional amplifiers, thus, have both $1 << A_{\min}$ and $-1 \ll C \ll \frac{-1}{A_{\min}}$ to keep both $|S_{\max}|$ small and $G_{\min}$ large. The result is a tiny sliver of acceptable values of (C, A) in parameter space as shown in Fig. S3, which is nonetheless compatible with real engineering components.

This simple model shares key features with the textile and biological architectures sketched above. There is no great complexity here, but what little there is (i.e., $C \neq 0$) is needed only for robustness, not minimal functionality in an idealized setting with no uncertainty. The functional parameters are an infinitely large (deconstrained) set but a very small, thin (constrained) fraction of all possibly parameter values, and thus, random circuits are vanishingly unlikely to be robust or functional. What is missing completely is any notion of dynamics, the addition of which, in the final case study, adds enormously to the nature of the constraints involved. Also, the functional parameter sets in Fig. S3 are convex, which is emphatically not true in general. Indeed, reparameterizing problems to make them convex is the heart of research in robust, distributed, and nonlinear control. Controllers are almost never convex in natural coordinates. This trivial math model also illustrates that complexity is hidden in normal function. One cannot tell from the outside what exactly is going on inside unless one knows the architecture and probes it for robustness with perturbations. This finding seems true of biology as well.

Simple Motor Control Example

For the last case study, we will consider a simple feedback motor control experiment that can easily be explored without specialized equipment, and it illustrates additional important constraints. The cartoon in Fig. S4 depicts the basic setup of a stabilization problem. The component constraints are that a mass m at location y on top of an inverted pendulum of length l is controlled by a muscle force u acting on a hand at position x, which is assumed to have effective mass M. The system constraint (and the main experimental problem) is that the hand must be controlled in such a way as to stabilize the up pendulum around $\theta = 0$ using the eyes to see the location y of the pendulum's top. This experiment is a standard experiment in human motor control and undergraduate engineering, replacing the hand with an actuated cart, but for our purposes, it illustrates important concepts that can be formalized and made rigorous mathematically.

Simple variations in the constraints can yield radically different properties. If the pendulum is sufficiently long, then it is easy to learn to stabilize the up position, but if too short, it is impossible

for humans (although robots can be built that will outperform humans in this simple task). Eyes closed, sensing only through contact with the hand, is a more severe component constraint, and it is apparently impossible for humans to stabilize with any length pendulum. Finally, the down position is naturally stable and comparably easy to control. The easiest experimental demonstrations of these extreme differences are obtained with pendulums of widely varying lengths and tip masses, but they can also be performed with a standard mechanical extendable pointer.

These three cases are summarized in the cartoon in Fig. S5 in a way that can be made rigorous and precise using robust control theory (see below for details). Here, we will summarize the results and explain their significance for constraints that deconstrain. The idea in Fig. S5 is that there are hard constraints that depend on the structure of the interconnection (up vs. down and eyes open vs. closed). To make Fig. S5 quantitative, fragility is defined in terms of the net sensitivity of the closed-loop control system (see below). These constraints are in the form of tradeoffs between the fragility of the controlled system and the length l. The tradeoffs plotted in Fig. S5, however, apply only to the unavoidable part of the fragility that depends just on the dynamics of the pendulum and the measurement and actuation point. This portion of fragility is independent of any additional uncertainty (e.g., noise in nerves and muscles) that arises in the controller implementation. Robust control design must treat both sources of uncertainty, but this use of robust control provides additional insights that we expect will be crucial to understanding robustness in biology.

The theory then predicts that, with eyes closed, the up system is far too fragile to be stabilized by humans, whereas the down case is trivially stable without control, although there are still lower limits on achievable fragility in response to external disturbances. This theory also predicts that, for up/open and a sufficiently short length l, the system is also too fragile to be stabilized. Although exact prediction of that length depends on the noise and time delays within the human controller, the qualitative dependence summarized in Fig. S5 of the unavoidable fragility on up/down, open/closed, and length l allows for a variety of conclusions to be rigorously inferred. One conclusion is the obvious benefit to this control task of remotely sensing the tip using vision rather than relying on hand contact alone, independent of additional details. This perhaps intuitively obvious observation can, thus, be elegantly and rigorously formalized.

Another interesting prediction of the theory that can be tested experimentally is that the hand will tend to oscillate at lengths near the limit of control. There is no purpose to these oscillations per se, and they are simply side effects of the hard constraints on robust control (in contrast to myriad oscillatory phenomena with obvious benefits such as clocks, wheels, pistons, cell cycles, radio carriers, etc.) Furthermore, this increase in control effort observed as l is shortened is a universal telltale sign of system approaching breakdown, although in other systems, it may be manifested as loss of control variability because of actuator saturations. Similarly, the illusions described in the paper may similarly be side effects of the kind of robust control, revealing not simple flaws but the consequences of intrinsic tradeoffs when systems are pushed near limits.

Another immediate consequence of robust control theory is that the controller parameterization, no matter what the implementation substrate (human or robot), necessarily is large/thin as well as nonconvex. The technical details of what constitutes large/thin/nonconvex parameterizations are not simple, but the intuition is simple. The set of controllers that robustly stabilizes the up/open case is a vast and essentially infinite set of dynamical systems, but it is a vanishingly small subset of all controllers (just as with outfits vs. heaps). In other words, random controllers will almost surely not stabilize. Tuning parameters in a robotic controller without computer-aided design tools is essentially impossible. Furthermore, if two parameter sets each are robustly stabilizing, their mean may not be stabilizing, and this nonconvexity is another universal property of robust controllers. Indeed, this information is all very well-known, because a major element of robust control theory is constructing embeddings of controller parameters in higher-dimensional abstracted parameter spaces that are convex and thus, searchable. Thus, the (non) convexity of parameter spaces has been a subject of intense study and is entirely consistent with the other case studies in the paper.

Pendulum Details

The pendulum example in Fig. S4 would correspond in Fig. S1 to $r = 0$ and A modeling the combination of hand and pendulum, with C being the controller, including the eyes, brain, nervous system, and muscle actuator. The noise n is assumed to model all of the internal noises in the controller as well as any external disturbances (e.g., air currents) in the environment, but we will not aim for a detailed characterization. All of the signals are now functions of time, and A and C have dynamics that will be explored in more detail later. We will consider a simple model that is standard in undergraduate courses and laboratories, where the cart moves in a line and the pendulum moves in a plane. This example ignores all of the 3D motion that makes the problem only moderately more difficult to control but vastly more difficult to explain in any detail. This problem is seemingly unavoidable. Algorithms that underlie robust control scale well (polynomially) with problem size but are rarely solvable analytically. The kind of analytic theory that we will do here, helpful for gaining insight into the nature of feedback and dynamics, is only possible with extremely simplified examples.

The standard equations of motion for a cart and pendulum in a plane are (Eqs. **S7–S9**)

$$\begin{aligned} &(M+m)\ddot{x} + ml\left(\ddot{\theta}\cos\theta - \dot{\theta}^2\sin\theta\right) = u, \\ &\ddot{x}\cos\theta + l\ddot{\theta} \pm g\sin\theta = 0, \text{ and} \\ &y = x + l\sin\theta. \end{aligned} \qquad \textbf{[S7–S9]}$$

The force of gravity is g. The up position equations correspond to the $-g$ in the $\pm g$ term, and if the whole diagram is flipped upside down (and the pendulum is grasped lightly by the hand), then the down equations are for the $+g$ case. It is not easy in practice to actually confine motion to a plane; however, this simplification will not hinder our use of this model, because we will be proving hard bounds on achievable robustness, and full 3D motion would simply make things more difficult. The pendulum stick is assumed to lack mass, with all of the mass concentrated at the tip.

The system in Eqs. **S7–S9** can be linearized to get the transfer functions from the control input $u(t)$ to both the output $y(t)$ and the hand position $x(t)$ [with transforms $U(s)$, $Y(s)$, and $X(s)$]. If we denote the transfer functions from $U(s)$ to $Y(s)$ and $X(s)$ as $A_Y(s)$ and $A_X(s)$, then we can solve analytically for (Eq. **S10**)

$$A_X(s) = \frac{ls^2 \pm g}{D(s)}\; A_Y(s) = \frac{\pm g}{D(s)}\; D(s) = s^2\left[Mls^2 \pm (M+m)g\right], \qquad \textbf{[S10]}$$

where $-$ and $+$ are for up and down, respectively. Note that, in the up case, both transfer functions are unstable, with a pole $P > 0$ that solves $D(p) = 0$, and the X transfer function has a zero $z > 0$ with (Eq. **S11**)

$$p = z\sqrt{1+r} = \sqrt{\frac{g}{l}(1+r)},\; z = \sqrt{\frac{g}{l}},\; r = \frac{m}{M}. \qquad \textbf{[S11]}$$

A standard control theory result is that the sensitivity function $S(s)$ in Eq. **S4** has the property that, if C is stabilizing, then

$S(p) = 0$, because $S(s) = \dfrac{1}{1 - C(s)A_Y(s)}$ and $A_Y(p) = \infty$. This result can be combined with standard results in complex analysis to show that, for all stabilizing controllers C (S12),

$$\frac{1}{\pi}\int_0^\infty \ln|S(j\omega)|d\omega \geq p > 0. \qquad \textbf{[S12]}$$

Here, $\ln|S(j\omega)|$ is the natural logarithm of $|S(j\omega)|$, which measures the amplification from the noise n to the output y at frequency ω. Ideally, $|S(j\omega)|$ is small, or equivalently, $\ln|S(j\omega)|$ is negative for a wide range of frequencies; however, from expression S12, the total integral of $\ln|S(j\omega)|$ is not only positive but is bounded below by p. Note also that p scales inversely with $\sqrt{l}$, and therefore, as $l \to 0$, the instability and the lower bound has $p \to \infty$. Intuitively, the pendulum eventually becomes too unstable and the feedback system becomes too fragile for a human controller to stabilize it. Thus, expression S12 is a hard constraint on the achievable robustness of the system and is depicted in Fig. S5, where fragility is now defined as the left hand side integral in expression S12.

This constraint also suggests what must happen on the way to instability, because as the length l is reduced, $\ln|S(j\omega)|$ will necessarily have at least one peak at some frequency ω that corresponds to oscillations in $y(t)$ [and also, $x(t)$]. This finding can be verified experimentally by finding the shortest length l that can be stabilized and noticing that the hand tends to oscillate with a period that corresponds roughly to the speed of response of the human nervous system. There is no purpose per se to these oscillations, which are simply unavoidable side effects of the stabilization problem and hard constraints. This finding seems to be a common source of confusion in biology, where purpose and meaning are sought for oscillatory or variable dynamics that are likely to be simply the result of hard constraints, partly because there are so many examples in biology and technology where oscillations are functional.

The down position has no instability even for small lengths, where any oscillations will be primarily because of the natural frequency of the pendulum, a rather different mechanism than in the up case. The constraint now becomes simply (S13)

$$\frac{1}{\pi}\int_0^\infty \ln|S(j\omega)|d\omega \geq 0, \qquad \textbf{[S13]}$$

which is still interesting, because it says that the total reduction in sensitivity over all frequencies is zero. Thus, all noise rejection must be matched by an equal amount of noise amplification. This finding is arguably the most important general constraint on feedback control, and it goes back to Bode in the 1940s (refs. 1–5 have related results). This finding will manifest itself experimentally in controlling the down pendulum, which is difficult when simultaneously controlling rapid hand and tip movement, but this finding also will depend somewhat on length and mass.

Perhaps the most interesting constraints occur when the eyes are closed and the only sensing occurs through the hand position (and also, the force of the pendulum on the hand). This constraint is most severe in the up case, which experimentally seems to be impossible to stabilize for any pendulum length. With the hand position x as the controlled output, the system $A_X(s)$ has not only an unstable pole at $p > 0$ but also a zero $z > 0$ as shown above. In contrast, the transfer function $A_Y(s)$ to y has no zeros and thus, is only constrained by expression S12. The zero $z > 0$ causes the constraint to strengthen to (Eq. S14)

$$\frac{1}{\pi}\int_0^\infty \ln|S(j\omega)|\left(\frac{z}{z^2+\omega^2}\right)d\omega \geq \ln\left|\frac{z+p}{z-p}\right| = \frac{\sqrt{1+r}+1}{\sqrt{1+r}-1} = \frac{\sqrt{M+m}+\sqrt{M}}{\sqrt{M+m}-\sqrt{M}}, \qquad \textbf{[S14]}$$

which is more complicated and harder to interpret but much more severe than expression S12. The system is simply inherently too fragile to be stabilized by a human controller, and even automatic cart–pendulum experiments require at least a sensor of θ to stabilize.

All of the constraints on robustness here are in the form of hard limits on the achievable sensitivity $|S(j\omega)|$ that can be derived from but do not trivially reduce to the other constraints on components, the system as a whole, and the protocols of interconnection. The term emergent has been so overused and misused as to become almost meaningless, but if we want to recover some useful meaning, emergent constraint could be taken to be just such a nontrivial derived constraint.

1. Chandra F, Buzi G, Doyle JC (2011) Glycolytic oscillations and limits on robust efficiency. *Science* 333:187–192.
2. Doyle JC, Francis BA, Tannenbaum A (1992) *Feedback Control Theory* (Macmillan, New York).
3. Zhou K, Doyle JC, Glover K (1996) *Robust and Optimal Control* (Prentice Hall, Englewood Cliffs, NJ).
4. Csete ME, Doyle JC (2002) Reverse engineering of biological complexity. *Science* 295: 1664–1669.
5. Doyle JC, Csete ME (2005) Motifs, control, and stability. *PLoS Biol* 3:e392.

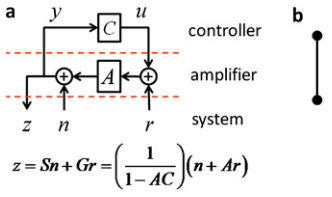

Fig. S1. (*A*) Block diagram of a minimal feedback amplifier circuit with reference input r, output z, noise n, measurement y, control u, amplifier A, and controller C. The system constraints are on r, z, and n in terms of $z = Gr + Sn$, which is implemented with amplifier and controller layers. The diagram is a visualization of the equations and usually would be different from a schematic of physical signals and connections. (*B*) A unipartite labeled graph model of the same system.

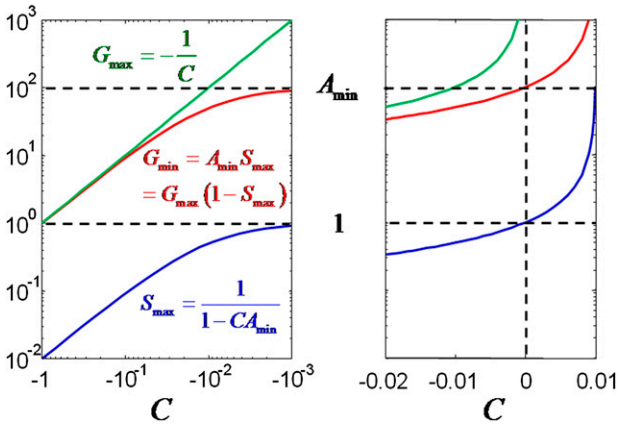

Fig. S2. System performance G_{max} (green), G_{min} (red), and S_{max} (blue) as a function of component values C for $A_{min} = 100$. *Left* has logarithmic units for C. In *Right*, note that, for positive feedback $C > 0$, $S_{max} > 1$ and noise is amplified.

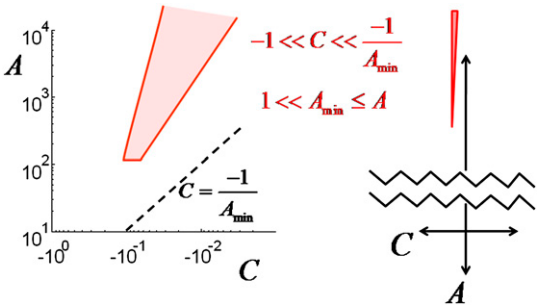

Fig. S3. Plot of the thin sliver $1 << A_{min} \leq A$ and $-1 << C << \frac{-1}{A_{min}}$ of functional parameter values for A and C in the space of all parameter values. *Left* is a log–log plot of a region blown up around the functional parameter values in red. *Right* is linear scaling, also focusing on the (red) functional region.

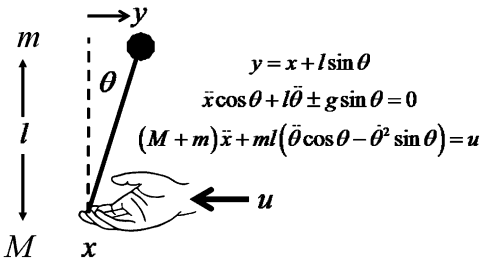

Fig. S4. Cartoon of the idealized stabilization problem with mass m at location y on top of an inverted (massless) pendulum of length l controlled by a control force u acting on the hand at position x and assumed to have effective mass M. The force of gravity is g. The up position equations correspond to the $-g$ in the $\pm g$ term, and if the whole diagram is flipped upside down (and the pendulum is grasped lightly by the hand), then the equations are for the $+g$ case. The best experimental results are obtained with pendulums of widely varying lengths and masses, but they can also be performed with a standard mechanical extendable pointer (details in the text).

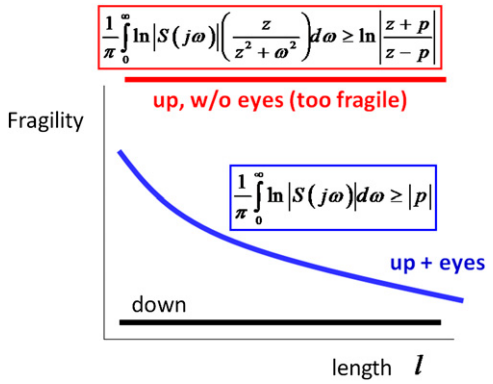

Fig. S5. Plots of hard constraint showing the net fragility as a function of the length of the pendulum and the structure of the controlled system. These cartoons can be made to be precise.