Explaining Neural Transitions through Resource Constraints*

Colin Klein
The Australian National University
colin.klein@anu.edu.au†

Abstract

One challenge in explaining neural evolution is the formal equivalence of a variety of different computational architectures. Well-known results show that various architectures, including neural networks with a single hidden layer and a nonlinear activation function, can be universal function approximators (Hornik et al., 1989). Why change? The answer must involve the intense competition for resources—including time, space, and energy— under which brains operate (Sterling and Laughlin, 2015). I argue that such explanations are ultimately an abstract species of resource explanation (Klein, 2018), which are distinct from but complementary to explanations in terms of mechanical parts. Resource explanations play an important role in computer science, one that is often under-appreciated by philosophers of neuroscience. As a case study, I show how the development of recurrence in neural networks can be favored when the increased complexity allows for more efficient use of existing resources. While resource competition drives the change itself, the development of recurrence creates shifts in the landscape of what is evolvable. The resulting framework suggests a mechanisms by which major neural transitions can occur, and shows why organisms on either side of a transition boundary may have very similar cognitive capacities but very different potential for evolving new capacities.

1 Introduction

Smith and Szathmáry (1997) proposed that we could understand the evolutionary history of organisms in part by thinking about a few *major transitions* in evolution. Major transitions included the jump from replicating molecules to

*Draft 1De1. Compiled August 21, 2021

†Thanks to Andrew Barron, Rachael Brown, and Marta Halina for helpful feedback on an earlier draft. This work was supported by grant TWCF0539 from the Templeton World Charity Foundation.

cells, the rise of sexual reproduction, and the origin of multicellularity. Each represents a major change in volvability – that is, in the possible ways that organisms might evolve. Each is also something of a one-way ticket: increased complexity is difficult to wind back once it has become stabilized.

More recently, several authors have argued that we might also understand the evolution of nervous systems as a series of major transitions. Ginsburg and Jablonka (2019) argue that the shift from limited to unlimited associative learning represents something like a major transition, explicitly on the model of Smith and Szathmáry and their transition from limited to unlimited heredity. Birch et al. (2020) further develop this idea, with unlimited learning as a 'transition marker' for a suite of capacities including conscious experiences. Barron, Halina, and Klein (draft ms) suggest that transitions might be understood in terms of changes in information flow in nervous systems, positing centralization, recurrence, and lamination as key major transitions. McShea and Simpson (2011) note that while changes in information flow are part of Smith and Szathmáry's discussion, they remain under-theorized and likely more important as organisms become more complex.

Yet while major transitions represent a powerful tool for thinking about how organisms and the space of evolvability might change over time, there are still considerable questions about how and why particular transitions occur. In the case of brains the question is relatively pressing, as it seems that even very simple brains possess considerable computational power. To put the problem starkly (and hint at the coming answer): there is a well-known result from computer science that suggests that neural networks with a single hidden layer and a nonlinear activation function are universal function approximators (Hornik et al., 1989). Anything we do with a complex, richly structured brain, then, could be done by a simple neural network of appropriate size. Why not just get bigger?

To make sense of a transition, we must make sense of why a transition to a new and more complex form of organization is favored at some time, given that the interesting benefits of evolvability come at some later time. Smith and Szathmáry note that "We cannot hope to explain these transitions in terms of the ultimate benefits they conferred" (Smith and Szathmáry, 1997, 8). The driver of evolution is usually immediate reproductive advantage.[1] In the case of brains, furthermore, it seems that merely making a new function—evolving a new sense organ, or a new processing trick, or any bread-and-butter evolutionary improvement—does not obviously get one any closer to a major transition. A transition is more than the aggregation of individually useful functions: it is a change in which functions are possible in the first place.

[1]Whether this is true of *all* changes in evolvability is something of a contested question—see Pigliucci (2008) for a review of the conceptual landscape. I follow Pigliucci's conclusion that the sense of evolvability at issue in major transitions is one that can only be selected for indirectly. In this paper I take no particular stance on whether evolvability also requires substantial developmental changes; see Brown (2014) for a helpful discussion of this issue. For what it's worth, I am also assuming that simple saltationist models, which assume a large *de novo* jump, are off the table

So the fan of major transitions in neural evolution would appear to face something of a mild puzzle. Transitions appear to have occurred and been important, and yet it is not obvious why or how they might occur. Now, this is obviously the sort of thing that should have a solution. As with most evolutionary puzzles, the key is finding something that might generate the necessary fitness gradient. In what follows, I will sketch an argument that a major driver of transitions could be general resource constraints. In section 2 I sketch a brief theory of explanation by resource constraint. In section 3 I then show how a transition might occur using purely resource considerations. I conclude in section 4 with a return to universal approximator theorems and the role of resource thinking on constraining our explanatory approach to neural evolution more generally.

2 Resource explanations

Resource explanations are, broadly speaking, a kind of mechanistic explanation. Mechanistic explanations decompose entities into parts, the coordinated activity of which explains some characteristic activity of the whole (Craver, 2007; Bechtel and Richardson, 2010). When we break down a mechanism into spatiotemporal parts, however, we find that these often come in one of two different flavors. To explain how an internal combustion engine produces rotary motion, we have to talk about pistons, valves, and injectors on the one hand, and gasoline and air and oil on the other. These play different roles within the broader explanation, and need slightly different treatment.

Following Klein (2018), I will distinguish between *mechanical parts* and *resources*. Very roughly speaking, this corresponds to an agent-patient distinction: mechanical parts do things (to other parts or to resources), while resources have things done to them. We can sharpen up that characterization along four different criteria.

First, mechanical parts tend to *persist* over the timescale of the explanation, while resources are often transformed. Gasoline and air are mixed and burned; the pistons and valves stay the same throughout. Note that transformation need not be dramatic or irreversible: cool oil is fed into the engine block, and is warmed as it removes heat. It is later cooled, and the cycle continues. Second, mechanical parts tend to be *individual* whereas resources are often *aggregate*. There are four pistons, and each of them needs to do the right thing at the right time. By contrast, gasoline is a mass that is broken into smaller bits as needed. While continuous resources are the cleanest cases, we can have discrete resources as well. A youth soccer team needs so many oranges at half time: the individual identity of the oranges is not important, only that there are enough for everybody. Third, mechanical parts are often *realization-indifferent* whereas resources are often *realization-sensitive*. Spark plugs and valves are classic functionalist examples: for the purposes of explanation nobody cares that much about what they are made of, so long as they are made of something that can do the job. Gasoline, by contrast, is not fungible: put in diesel and

the engine won't work.

Note that the mechanical part/resource distinction—and therefore the satisfaction of the first three criteria—is often explanation-relative. If I care how my car stops, brake pads are mechanical parts: persistent, individually important, and functional. If I'm managing a racing team, I may go through so many brake pads that I treat them as a consumable resource. Whole mechanisms in one context can be mere resources in another: the jeep is a complex whole to the mechanic, matériel to the quartermaster. The point of distinguishing mechanical parts and resources is thus not to draw a firm metaphysical line in the world, but to emphasize the different roles that different spatiotemporal parts of a mechanism can play within the same explanation.

Fourth and finally, mechanical parts are usually *causally conservative* whereas resources are *causally promiscuous*. Each valve has a fixed role and interacts with a few different things in predictable ways. Indeed, an important rule of thumb in engineering is to keep systems modular if possible: that is, to minimize the ways in which mechanical parts interact (Simon, 1996; Calcott, 2014). By contrast, resources are often used by multiple different processes at once. In most cars, engine oil both lubricates and cools the engine. The functions can interact: if the oil gets too hot, it ceases to be a good lubricant.

Indeed, in many domains there is a familiar possibility of *resource competition* with attendant *resource starvation* as an explanation of failures. It takes so much rangeland to support so many head of cattle. Why? Any individual animal would need much less, but the fact that they are all competing for the *same* grass means that starvation threatens on a smaller plot. Conversely, there are many engineering problems that arise from the need for *resource management*. Electrical grids would be simple if there was one producer and one consumer. Delivering energy to many different households with different patterns of demand requires considerable infrastructure just to make sure that everyone gets what they need.

I have mostly spoken of concrete resources. However, the point can easily be generalized to more abstract resources as well. Economics is all about the interaction between various concrete resources (pigs, whiskey, fireworks) and various abstract ones (money, futures, derivatives). A key set of abstract resources are found in computational theory. In particular, computational complexity theory studies how different algorithms require different amounts of time, memory, processor cycles, bandwidth, and so on (Aaronson, 2015). Sometimes these resources trade off against one another: we might cache results from a computation to save time at the expense of space. They also enter into explanations that involve resource competition and resource starvation. My poor choice of a sorting algorithm means that my computer used too much memory, which explains why the operating system crashed. Both would have been fine on their own, but the fact that they were competing for the same, causally promiscuous, pool of memory means that they couldn't co-exist.

3 Transition and elaboration

Return to the evolution of brains. Brains—and nervous systems more broadly—are incredible resource hogs. Raichle and Mintun (2006, 467) estimate that the human brain uses about 20% of our total energy expenditure despite being only about 2% of body weight. Furthermore, the majority of that is a standing cost: it is paid whenever we are awake, regardless of what we are doing. In Sterling and Laughlin (2015)'s survey of overarching principles of neural design, they show that resource constraints shape even the most basic nervous systems. Some of these constraints also have important nonlinearities. Notably, the both the energetic and volumetric costs of sending information rise disproportionately at the rate increases (Sterling and Laughlin, 2015, 54).

The basic question about transitions, recall, was why a change in information-flow might be favored, given that the obvious benefits of such a change to evolvabiilty of new functions don't arise until after the transition. Put in the language of the previous section, transitions may well allow for the evolution of new mechanical parts, but the possibility of new parts cannot be the reason why the transition occurs. An obvious place to look for the answer is instead with resource constraints. Furthermore, as I'll argue, it is possible for a transition favored on resource grounds alone to facilitate new patterns of information flow and thus new functional parts down the line.

Here is an example of the sort thing I have in mind, using an example from artificial neural networks. Consider a recurrent neural network containing s neurons and requiring t timesteps to calculate some function f. There is a well-known result that shows that this network can be 'unrolled' into into a purely feedfoward network, which computes f with containing t layers and st neurons.[2] Assume that the timescale is comparable in each case (that is, that recurrent loops take the same amount of time as additional layers), so that the recurrent and unrolled network also take the same amount of time to compute f.

Now, flip this picture around. Suppose an organism with purely feedforward connectivity calculates some f by circuit N, but that f could be calculated by a recurrent network N', such that N is the unrolled version of N'. *Ex hypothesi*, both circuits calculate the same function in the same amount of time. However, if N has several layers, the shift to N' might come at a substantial energetic savings, because N' would use $s(t-1)$ fewer neurons.

[2] I use the formulation found in Šíma and Orponen (2003, p2746) Very literal readings of this should be taken with a grain of salt. The result is typically attributed to Savage (1972), with Goldschlager and Parberry (1986, p56) the first to dub it 'unrolling'. Both the Savage and the Golschlager & Parberry articles concern networks of traditional Boolean gates, however, and these do not need to be trained. The fact that useful trainable recurrent nets use specialized gates, higher-order weight functions, or other similar departures from simple perceptron models suggest the need for more nuance. That said, there are a number of intuitive presentations of recurrent neural networks in terms of unrolling to purely feedforward neural nets, and the idea that a recurrent net can be translated into a feedforward net with a space penalty roughly *proportional* to time should be uncontroversial.

In fact, the tradeoff is a bit more complex. Artificial recurrent networks are harder to train: simple backpropagation faces a problem of vanishing or exploding error gradients.[3] Recurrence in biological systems faces an analogous problem, one assumes, due to the inherent instability of excitatory feedback loops. Long-range feedback might also mean more long-range wiring costs, which are themselves substantial resource drain (Sterling and Laughlin, 2015) and one that brains seek to minimize (Cherniak et al., 2004). So there are additional steps needed to make a recurrent network stable and trainable. Let's assume that these costs scale with the number of neurons by some constant factor c. Then we might expect an evolutionary transition from N to N' to be favored, on resource grounds alone, just when $st > sc$.

In other words, we should not expect recurrence to evolve and stabilize when the additional costs of stabilizing a recurrent network is more than the cost of the additional layers needed by a feedforward network. Furthermore, since these tradeoffs are vague and approximate, if recurrence *does* occur, we should expect there to be some point where both N and N' are both live options—that is, N' does not have an obvious advantage over N, despite added complexity.

Or, to be more precise, N' does not have an advantage *with respect to computing* f. However, the transition to a recurrent network might bring benefits for computing *other* functions. For N' can compute functions that take longer than t without having to add additional hardware. What N' can do with time, N must do with space—and time is often cheaper than space. That is, if an organism has the leisure to run a function for longer, it gets the benefit of recurrence for minimal additional metabolic cost. Adding neurons and wiring, by contrast, adds a substantial fixed cost. Furthermore, the choice of whether to run an algorithm longer can be made on the fly, whereas making a bigger brain usually requires developmental changes.

For a concrete example of the sort of algorithms that benefit from more time, one might consider what Zilberstein and Russell (1996) call 'anytime algorithms'. These approximate a certain function and do a better job the more time they are given, and hence "allow computation time to be traded for decision quality" (Zilberstein and Russell, 1996, 181). Some algorithms of this sort, like Newton's method for finding roots, are well known. However, there are interruptible anytime versions of algorithms for many problems faced by real-time control, like the traveling salesman problem (Zilberstein and Russell, 1996, 190ff).

My claim is that a network like N' might get the benefits of these algorithms effectively for free, while N has no obvious way to perform additional iterations aside from adding more layers (and thereby commit more resources). That in turn opens up the possibility of real functional change, by making possible

[3] As Schmidhuber (2015, 93ff) notes, this problem was known by the late 1980s, received formal expression by Hochreiter's 1991 PhD thesis, and was the focus of intense research for nearly 20 years before recurrent neural networks were competitive at major contests. The advances required to make RNNs competitive were not simply increases in computational power (thought that helped), but also fundamental algorithmic advances. Once RNNs were feasible, however, they rapidly came to dominate at many tasks—reflecting the argument of this paper in miniature.

algorithms that would be too costly to be useful in simple feedforward networks.

This is all a how-possibly explanation, of course, and an abstract one at that. The point is not to make claims about an actual transition, but rather to show how resource considerations *alone* could support a neural transition. The explanatory pattern above suggests that the transition itself might be favored on pure resource competition grounds. The same function f is computed in the same amount of time before and after the transition. However, the *consequence* of the transition may well be the evolvability of more complex, more efficient, or more useful functions—functions that the original network may resist evolving precisely because of the added cost.

4 Conclusion

In setting up the problem of transitions, I noted the universal approximator theorem and suggested that anything a complex brain does could also be done by a sufficiently large simple brain. Why not just make a simple big brain then? Answer: because big brains are costly, and at some point the benefits of simplicity are outweighed by those costs. The transition to a more complex pattern of information flow may solve a proximate resource problem. In doing so, however, it may open up the possibility of evolving more sophisticated and more complex functions. A resource-driven transition might change the pattern of evolvability more generally, then, just as the major transitions framework predicts.

I note that while resource pressures are particularly pressing for brains, they have also been cited as drivers for more basic transitions as well. Knoll and Hewitt (2011) have an excellent discussion about how many features of multicellularity are driven by the limitations of passive diffusion as a transport mechanism. Once a multicellular organism gets large enough, it cannot rely on simple gradients of nutrients from the outside to the inside. While there are short-term fixes, a common pathway seems to lead to developmental changes, which ultimately lead to the specialization of function that is a common feature of increasing complexity (Calcott, 2011). Again, this strikes me the sort of transition that is fundamentally driven by pressures on resource management and resource allocation.

Indeed, while I have focused on the efficiencies to be gained by a transition from purely feedforward to recurrent networks, I suggested that there are costs to be borne as well. In many engineered systems, problems of resource competition are solved by systems dedicated to resource management: if everyone in the house wants to stream a movie at once, a good router will try to balance the load to make sure that no one person saturates the connection. As systems get more complex, more and more effort must be devoted to resource management, including higher-order problems of resource management. Money helps solve the problems arising from the allocation of scarce concrete resources. Banks help

solve problems that arise from managing large quantities of money. Regulators manage banks. And so on. Each level of this hierarchy uses some of the very resources they manage (bankers like to get paid), which in turn creates more complex resource management problems.

We should not expect the brain to be different. The transition to more complex brains comes with increasing pressure on resource management. Indeed, in complex brains like ours, I suspect this becomes a central preoccupation. Resource explanations of ever increasing complexity might therefore be the key to understanding major transitions in neural evolution.

References

Aaronson, S. (2015). Why philosophers should care about computational complexity. In Copeland, B. J., Posy, C., and Shagrir, O., editors, *Computability: Gödel, Turing, Church, and Beyond*, pages 261–327. MIT Press, Cambridge.

Bechtel, W. and Richardson, R. (2010). *Discovering complexity: Decomposition and localization as strategies in scientific research*. MIT Press, Cambridge.

Birch, J., Ginsburg, S., and Jablonka, E. (2020). Unlimited associative learning and the origins of consciousness: a primer and some predictions. *Biology & philosophy*, 35(6):1–23.

Brown, R. L. (2014). What evolvability really is. *British Journal for the Philosophy of Science*, 65(3):549–572.

Calcott, B. (2011). Alternative patterns of explanation for major transitions. In Calcott and Sterelny (2011), pages 35–52.

Calcott, B. (2014). Engineering and evolvability. *Biology & Philosophy*, 29(3):293–313.

Calcott, B. and Sterelny, K., editors (2011). *The major transitions in evolution revisited*. The MIT Press.

Cherniak, C., Mokhtarzada, Z., Rodriguez-Esteban, R., and Changizi, K. (2004). Global optimization of cerebral cortex layout. *Proceedings of the National Academy of Sciences*, 101(4):1081–1086.

Craver, C. (2007). *Explaining the brain*. Oxford University Press, New York.

Ginsburg, S. and Jablonka, E. (2019). *The Evolution of the Sensitive Soul : Learning and the Origins of Consciousness*. MIT Press, Cambridge.

Goldschlager, L. M. and Parberry, I. (1986). On the construction of parallel computers from various bases of Boolean functions. *Theoretical Computer Science*, 43:43–58.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.

Klein, C. (2018). Mechanisms, resources, and background conditions. *Biology & Philosophy*, 33(36):1–14.

Knoll, A. H. and Hewitt, D. (2011). Phylogenetic, functional, and geological perspectives on complex multicellularity. In Calcott and Sterelny (2011).

McShea, D. W. and Simpson, C. (2011). The miscellaneous transitions in evolution. In Calcott and Sterelny (2011).

Pigliucci, M. (2008). Is evolvability evolvable? *Nature Reviews Genetics*, 9(1):75–82.

Raichle, M. E. and Mintun, M. A. (2006). Brain work and brain imaging. *Annual Review of Neuroscience*, 29:449–476.

Savage, J. E. (1972). Computational work and time on finite machines. *Journal of the ACM (JACM)*, 19(4):660–674.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Šíma, J. and Orponen, P. (2003). General-purpose computation with neural networks: A survey of complexity theoretic results. *Neural Computation*, 15(12):2727–2778.

Simon, H. A. (1996). *The Sciences of the Artificial*. MIT Press, Cambridge, 3rd edition.

Smith, J. M. and Szathmáry, E. (1997). *The major transitions in evolution*. Oxford University Press, New York.

Sterling, P. and Laughlin, S. (2015). *Principles of neural design*. MIT Press, Cambridge.

Zilberstein, S. and Russell, S. (1996). Optimal composition of real-time systems. *Artificial Intelligence*, 82(1-2):181–213.