# Watching Evolution Happen: Reinforcement Learning as a Workbench for the Law of Increasing Functional Information

**Albert Jan van Hoek**

**a.j.van.hoek@gmail.com**

All views and ideas in this paper are my own, developed outside the scope of my employment, and should not be attributed to my employer.

## Abstract

In 2023, Wong, Hazen, and colleagues proposed a 'missing law' of nature: the Law of Increasing Functional Information. This law states that when a system's many configurations are subjected to selection for one or more functions, functional information will accumulate. The authors demonstrated this across minerals, stellar nucleosynthesis, and biology. Here I argue that reinforcement learning (RL) represents an ideal workbench for observing this law in action—perhaps the first time in history we can watch functional information accumulate at human-observable timescales, with full experimental control over all parameters. To see RL this way requires viewing it at a higher level of abstraction. Chen and colleagues recently provided exactly this abstraction by mapping RL dynamics onto electronic circuit principles. Recognizing this abstraction as identical to the evidence-based decision loops I construct as a health economist, I saw the underlying pattern: a process operating on networks that generates emergence through random interaction, retains functional configurations through feedback, and ratchets upward as retained structures become available for new combinations. This process is the Law of Increasing Functional Information grounded in network structure. RL is special not because it alone follows this law, but because we stand outside it with full experimental control, observing a universal process that also contains us.

## Seeing Reinforcement Learning Abstractly

To recognize reinforcement learning as a demonstration of the Law of Increasing Functional Information, we first need to see RL at the right level of abstraction. Viewed as code—parameters, gradients, loss functions—the connection is obscure. Viewed as a process, the connection becomes exact.

Chen and colleagues recently provided this abstraction. In their Electronic Circuit Principles framework, they map RL dynamics onto circuit elements: Their framework achieves strong empirical predictions validating that this higher-order view captures something real about how RL systems behave.

What struck me reading Chen et al. was not the circuit metaphor itself, but its familiarity. The process they described—gathering information, synthesizing it into a model, subjecting the model to constraints, iterating toward functional output—was identical to what I do as a health economist building cost-effectiveness models. In cost-effectiveness analysis, I gather data from published studies and clinical trials, synthesize this into a model structure, subject the model to constraints (review and quality assurance), and iteratively refine both model and data until the output serves its function: informing resource allocation decisions.

The same process. In neural networks and in science.

This recognition matters because evidence-based decision making—the loop of gathering evidence, building models, testing against reality, refining—is simply what science is. And science is simply what careful thinking is. The abstraction Chen et al. provided for RL is not a metaphor borrowed from elsewhere; it is the same process appearing across domains because it is how functional information accumulates everywhere.

## The Law of Increasing Functional Information

Wong and Hazen's law makes a precise claim: the functional information of a system will increase if many different configurations of the system are subjected to selection for one or more functions. Three conditions are necessary and sufficient: a vast configuration space, selection pressure based on function, and iterative operation over time.

Functional information measures how rare working configurations are within total possibility space. If only one in a million configurations performs a given function, that configuration contains more functional information than if half of all configurations work equally well. As selection operates, functional configurations persist while others disappear. The population concentrates in increasingly rare regions of configuration space—regions that, by definition, contain more functional information.

Wong and Hazen demonstrated this across mineralogy, stellar nucleosynthesis and biological evolution.

But their formulation leaves something implicit: what generates these conditions? The answer I propose is network structure.

## Network as Substrate

I would suggest that configuration space arises from network topology—the combinatorial possibilities of how nodes can be connected and weighted. Selection for function operates on network-level outputs—function is an emergent property of collective node behavior, not a property of individual nodes. Iteration is possible because networks permit incremental state changes without systemic collapse.

This matters because the law is not merely analogous across minerals, organisms, and neural networks. These systems share network structure, and network structure is what makes the law operate. Minerals are networks of atomic bonds. Organisms are networks of cells, proteins, and genes. Neural networks are, obviously, networks.

The mechanism of selection operates at the node-network interface. Each node's state contributes to network behavior. Network behavior, evaluated against function, feeds back to node states. Selection preserves configurations where node-level states produce network-level function.

In RL terms: nodes are parameters, the network is model architecture, selection is the reward or loss signal (a network-level output), and gradient descent is the mechanism translating network-level selection to node-level updates.

## The Loop and the Ratchet

The process that accumulates functional information is both a loop and a ratchet. It cycles, and it builds.

The sequence:

First, network—the substrate of interconnected elements where the process occurs.

Second, random interactions—elements interact within the network, exploring configuration space through variation.

Third, emergence—some interactions produce functional configurations. Capabilities arise from arrangements that no element possesses alone. This is not mysterious; it is what functional information describes. A configuration 'works' when the arrangement of elements produces function that emerges from their combination.

Fourth, feedback—the emergent function is evaluated. Critically, this feedback comes from the network itself. The network is both substrate and evaluator – in a wide sense – the reason a species stays is because of feedback from within the ecosystem. There is no external judge; the network's own dynamics determine what counts as functional.

Fifth, retention—functional configurations are preserved by the network. This is the ratchet mechanism. Without retention, the process would cycle endlessly without accumulation. Retained emergence is locked in, stabilized within network structure.

Sixth, enrichment—the retained emergence becomes available as a unit in the network. What emerged and was retained can now participate in new interactions.

Seventh, return—but now the network includes more structure. Random interactions occur among elements that include previously retained emergences. New combinations become possible that were not possible before.

This dual nature—loop and ratchet—explains how functional information accumulates rather than merely cycling. Each iteration builds on the last. Retained emergences become building blocks for the next round. The system doesn't start over; it starts higher.

In RL: early training produces basic features that get locked into weights. Mid training uses those features as building blocks for higher-order patterns. Late training produces complex capabilities from combinations of already-retained structures. The loss curve descending is functional information ascending.

## The Paradox of a Universal Law

Here we encounter something philosophically peculiar. If the Law of Increasing Functional Information is truly universal—if it governs everything that exists—then there is no outside from which to validate it. Every potential comparison is itself inside the system. When we say 'RL is like biology,' we are not providing external proof; biology follows the same law. All examples are internal references.

This is not a weakness of the law but a feature of its universality. A law that governs all systems cannot be validated by comparison to systems outside its scope. There are no such systems.

This creates the odd situation where the law seems to explain everything while being impossible to test against a control condition. But impossibility of external validation is precisely what universality means.

What we can do is observe the law operating in systems where we have experimental control—systems where we are, for practical purposes, outside.

## Two Levels of System

This requires distinguishing two levels:

The first level is RL as a controlled system. We define the loss function. We can change it, observe it, manipulate it. The feedback loop is transparent to us. We stand outside this system, adjusting parameters, observing outcomes, running controlled experiments. This is the petri dish, the workbench.

The second level is existence as the containing system. The computer running RL exists within larger networks. We exist within larger networks. These networks have their own feedback, their own retention mechanisms. We don't control this loss function in a similar way—we don't even fully see it at times. We are nodes in this system, not observers of it. The 'function' being selected for operates on us, not merely through us.

In RL, we design the function, watch the emergence, control the retention.

In existence, the function is given. We are the emergence. Retention operates on us.

## Why Reinforcement Learning Is Special

RL is not special because it is the only place the Law of Increasing Functional Information operates. The law operates everywhere. RL is special because it is a system where we are external to it, where we control the parameters, where we can observe the full loop, and where timescales are human-accessible.

This combination is unprecedented. Mineral evolution operates over billions of years; we infer the law from geological records. Biological evolution operates over millions of generations; we infer the law from phylogenetic trees and fossils. Even laboratory evolution in bacteria requires generations beyond direct observation of mechanism.

In RL, a single training run involves thousands of selection events across millions of parameter updates, completed in weeks or months. We can pause training, measure functional information (via performance metrics), resume training, measure again. We can compare identical starting conditions under different selection regimes. We can checkpoint intermediate states and examine exactly what configuration existed at each stage.

We are watching evolution happen. Not inferring it from traces. Watching it.

## The Observation

What do we observe?

We observe functional information accumulating. Models improve systematically through training. Performance metrics—accuracy, capability, coherence—increase as selection accumulates. The models become concentrated in rare, functional regions of parameter space: most possible parameter configurations produce garbage; only very specific configurations produce coherent output.

We observe the ratchet. Early capabilities become building blocks for later capabilities. Models don't re-learn basic features at each training step; they retain them and build upon them. The emergence is cumulative.

We observe emergence at thresholds. As training accumulates, models suddenly acquire capabilities they did not previously possess. A model that could not perform a task begins performing it—not gradually, but as a phase transition when accumulated structure crosses a threshold. This is precisely what the law predicts: concentration in functional regions enables new functions when density reaches critical points.

We observe the network dynamic. The feedback that drives selection is computed from network-level outputs. Individual parameters have no 'preference'; they are adjusted based on their contribution to collective behavior. The node-network dynamic—parameters serving model-level function—is exactly visible in the gradient descent mechanism.

## Implications

If RL demonstrates the Law of Increasing Functional Information in an artificial substrate with full experimental control, several implications follow.

First, the law is confirmed as substrate-independent. It is not a special feature of carbon chemistry or biological reproduction, but a consequence of network structure under selection. Silicon works. Code works. Anything with the right structure works.

Secondly, the loop-ratchet structure matters. Functional information accumulates because the process both cycles and retains. Understanding this dual nature clarifies how emergence builds on emergence, how complexity ratchets upward rather than merely fluctuating.

Third, the two-level distinction clarifies what kind of knowledge this is. We can experimentally validate the law in systems we control (Level 1), while recognizing that the same law operates on us in systems we do not control (Level 2). This is not a limitation; it is the epistemic situation any universal law creates.

## Conclusion

Wong, Hazen, and colleagues proposed a law that unifies evolutionary phenomena under a single principle: functional information increases when many configurations are subjected to selection for function. This paper has argued that reinforcement learning provides an ideal workbench for observing this law.

Seeing RL this way required a higher-order abstraction, which Chen et al. provided through their electronic circuit framework. Recognizing this abstraction as identical to evidence-based decision making in health economics—and therefore to science, and therefore to thinking—revealed the universality of the underlying process.

But Wong and Hazen's law is compatible with network thinking. Network structure generates configuration space, enables selection on emergent function, and permits iterative refinement.

The process is both loop and ratchet: random interactions produce emergence; feedback selects for function; retention preserves functional configurations within the network; retained emergences become available for new combinations. The ratchet explains accumulation.

A universal law cannot be validated externally; there is no external. But we can observe it operating in systems we control—systems where we stand outside, manipulating parameters, watching outcomes. RL is such a system. In RL we design the function, observe the emergence, control the retention. In existence, we are the emergence, and retention operates on us.

We built a workbench and the law appeared. Not because we programmed it in, but because the structural conditions—network, selection, iteration—made its operation inevitable. The Law of Increasing Functional Information, observed in reinforcement learning, reveals itself as what it always was: a law of networks under selection, operating wherever the structure exists to support it.

We are watching evolution happen. And in watching it, we begin to understand what evolution actually is – and it feels very personal.

I believe there are many relevant implications of these ideas, the process of emergence is a process of recombination and collaboration, which is perhaps a process missing in Darwinian evolution. I have started to explore these ideas in a GitHub project called "Evolution by Emergence." Please feel invited to review and collaborate on materials in this repository too.

## References


Wong, M.L., Cleland, C.E., Arend, D., et al. (2023). On the roles of function and selection in evolving systems. *Proceedings of the National Academy of Sciences*, 120(43), e2310223120.

Chen, Q., Qin, L., Liu, J., et al. (2025). Electronic Circuit Principles of Large Language Models. *arXiv preprint* arXiv:2502.03325v2.


Github repository: https://github.com/albertjanvanhoek/Evolution-by-Emergence