

Modularity, not integration, is the architectural basis of universal computation

Jose Marie Antonio Minoza^{1*}, Erika Fille T. Legara^{1,2} and Christopher P. Monterola²

¹Center for AI Research, Department of Education, Philippines.

²Asian Institute of Management, Makati, Philippines.

Abstract

Elementary Cellular Automata (ECA), despite their minimal rule set, are capable of reproducing a remarkably diverse range of naturally occurring patterns, from chaotic dynamics to structures supporting universal computation. While their forward behavior has been extensively cataloged, how such complexity arises from simple local rules remains an open problem. In this work, we approach ECA as a reverse-engineering problem; rather than analyzing rules directly, we infer their governing dynamics by evolving neural networks to learn and reproduce all 256 ECA rules. We implement NeuroEvolution of Augmenting Topologies (NEAT) to evolve rule-learning networks and quantify their functional information using five complementary metrics: structural information (I_S), behavioral information (I_B), effective information (I_E), integration (Φ), and transfer entropy (TE). Across 30 evolutionary runs per rule, we uncover a robust inverse relationship between structural complexity and integration. Class IV rules, most notably the Turing-complete Rule 110, require high structural complexity ($I_S \approx 222$ bits) yet exhibit near-zero integration ($\Phi \approx 0.007$), indicating highly modular internal organization. In contrast, Class III XOR-type rules are learned with minimal structure ($I_S \approx 83$ bits) but achieve maximal integration ($\Phi > 0.45$), reflecting tightly coupled computational dynamics. Variance analysis further reveals that highly integrated rules occupy constrained solution spaces with low architectural diversity, whereas modular, low- Φ rules admit a wide range of viable network realizations. These findings suggest that universal computation in ECA emerges not from high integration, but from modular compositionality. More broadly, our results demonstrate how reverse engineering via neuroevolution can expose latent organizational principles in simple dynamical systems, offering a data-driven refinement of the law of increasing functional information. Evolutionary accessibility, rather than raw complexity, favors modular architectures, shedding new light on how simple rules give rise to rich, naturalistic patterns.

Keywords: Elementary Cellular Automata, Functional Information, Neuroevolution, Wolfram Classes, Computational Complexity

Introduction

The recently proposed “law of increasing functional information” holds that evolving systems accumulate structural complexity through selection for function [1]. The framework unifies diverse evolutionary phenomena, from mineral diversity to protein evolution to technological innovation, under a common information-theoretic principle. But the law is silent on a crucial question: *how* is this complexity organized? Does functional information accrete as tightly integrated wholes, requiring precise coordination among components? Or does it assemble from loosely coupled modules that can be modified independently? The answer determines whether increasing complexity leads to evolutionary dead ends or continued adaptability; and, may explain a ubiquitous pattern in biological organization: modularity.

Modular architectures are pervasive across biological systems. Proteins fold into semi-independent domains connected by flexible linkers [2]. Genetic regulatory networks exhibit dense local connections but sparse long-range coupling [3]. Neural circuits segregate into specialized regions with limited inter-area integration [4].

Simon [5] proposed that evolution favors *nearly decomposable* systems because modular architectures can be assembled incrementally—piece by piece, with each piece tested and refined independently. Yet an alternative remains viable; perhaps, tight integration is the signature of powerful computation, and biological modularity reflects other constraints entirely. Distinguishing these possibilities requires a system, where computational function is precisely specified, and evolutionary outcomes can be measured across many independent trials.

Elementary cellular automata (ECA) provide such a system. Despite their minimal specification, where each rule is defined by just eight bits, ECA generate the full spectrum of dynamical behavior, from trivial fixed points to chaotic dynamics and universal computation [6]. Cook’s proof that Rule 110 is Turing-complete established that even the simplest deterministic systems can support arbitrary computation [7]. Crucially, ECA decouple computational power from apparent dynamical complexity: some rules produce visually complex patterns while remaining computationally simple, whereas others support universal computation through subtle combinations of localized structures. Because each rule’s function is exactly specified and the space of possible implementations is vast, ECA form an ideal testbed for examining how functional information scales with computational capacity.

Here, we use evolutionary search to probe the organization of functional information in all 256 elementary cellular automata. We evolve neural networks to learn each rule’s local update function, starting from minimal architectures and allowing structure to accumulate only when it improves performance [8]. Repeating this process across many independent evolutionary runs enables us to characterize not only the structural requirements of each computation, but also the diversity of viable implementations. This framework allows us to quantify multiple complementary aspects of

functional information, including structural complexity, causal influence, integration, and information flow [9, 10].

Using this approach, we show that computational universality does not arise from tightly integrated architectures. Instead, universal rules rely on modular organization, combining many relatively independent components. Moreover, we find that modular architectures admit a broader range of evolutionary solutions, whereas highly integrated architectures are comparatively constrained. Together, these results suggest that evolution accumulates not merely more functional information, but modularly organized information—structure that remains accessible to further adaptation.

We evolve neural network implementations of all 256 ECA rules using neuroevolution, then characterize the resulting architectures using complementary information-theoretic measures. Full methodological details are provided in Materials and Methods.

Results

Neural networks were evolved to learn the local update function of all 256 elementary cellular automata, yielding 7,680 converged networks across 30 independent evolutionary runs per rule. Networks achieving greater than 99.9% accuracy on all eight input configurations were included in the analysis.

Universal computation requires high structural complexity but low integration

The Aggregate Functional Information measures grouped by Wolfram class are shown in Table 1. Clear differences emerge across classes. Class IV rules require the greatest structural complexity; they exhibit the highest Structural Information and the largest number of hidden nodes, indicating that implementing universal computation demands deeper and more elaborate architectures. At the opposite extreme, Class I rules converge to minimal network structures, consistent with their trivial computational behavior.

Table 1 Aggregate Functional Information by Wolfram class. Values represent mean $\pm$ standard deviation across all rules in each class. Product indicates the aggregation function used by evolved networks.

Class	I_S (bits)	Hidden	Φ	T_E (bits)	Product (%)
I (24 rules)	166 ± 46	4.6 ± 1.3	0.015 ± 0.014	0.065 ± 0.024	97%
II (183 rules)	191 ± 41	5.3 ± 1.2	0.014 ± 0.021	0.080 ± 0.020	98%
III (43 rules)	192 ± 43	5.5 ± 1.1	0.068 ± 0.134	0.079 ± 0.015	99%
IV (6 rules)	208 ± 39	6.2 ± 1.3	0.017 ± 0.038	0.084 ± 0.009	100%

Structural complexity, however, does not correspond to functional integration. Despite higher Structural Information, Class IV rules exhibit consistently low integration values. By contrast, Class III rules show the highest mean integration across all

classes, even though their structural complexity is comparable to or lower than that of Class IV. This elevated integration is driven by a subset of algebraically irreducible rules rather than by computational universality itself.

Transfer entropy increases modestly from Class I to Class IV, reflecting stronger directional information flow in more complex dynamics. Unlike Structural Information, however, transfer entropy does not distinguish universal rules from chaotic or periodic ones. Together, these class-level results reveal a dissociation between structural complexity and functional integration, motivating closer examination of the architectures underlying these computations.

The results show that the number of hidden nodes required is determined more by a rule’s underlying logical structure than by the apparent complexity of its output. Even within the same class, representational demands vary widely: simple parity-based rules are learned efficiently, while rules with higher-order interactions require substantially more capacity. This explains why visually chaotic rules can differ sharply in neural complexity and highlights the limits of classifying rules solely by their observable behavior.

NEAT-evolved architectures make the modularity–integration contrast visible

The organizational differences suggested by the class-level analysis are directly visible in the evolved network architectures. Figure 1 shows representative networks evolved for four characteristic rules: Rule 30 (chaotic), Rule 90 (XOR), Rule 110 (Turing-complete), and Rule 150 (three-input XOR). All networks compute deterministic mappings from the same three-bit input to a single output, yet their internal organization differs markedly.

XOR-type rules, such as Rule 90 and Rule 150, evolve compact architectures in which multiple inputs must be combined jointly to determine the output. These networks are tightly coupled internally and exhibit high integration despite requiring relatively few hidden nodes. Their computation is algebraically irreducible, where no subset of inputs alone carries information about the output.

In contrast, networks evolved to implement Rule 110 are substantially larger and deeper, yet exhibit little functional integration. Their internal structure decomposes into loosely coupled components that process inputs in a largely independent manner. Despite supporting universal computation, these architectures do not rely on holistic integration, but instead assemble simpler subcomputations into a coherent whole.

Chaotic rules, such as Rule 30, illustrate a third pattern. Although Rule 30 produces highly irregular spatiotemporal dynamics, its evolved networks show neither high integration nor minimal structure. Instead, they exhibit strong directional information flow without tight coupling, underscoring that chaotic behavior alone does not imply either integration or computational universality.

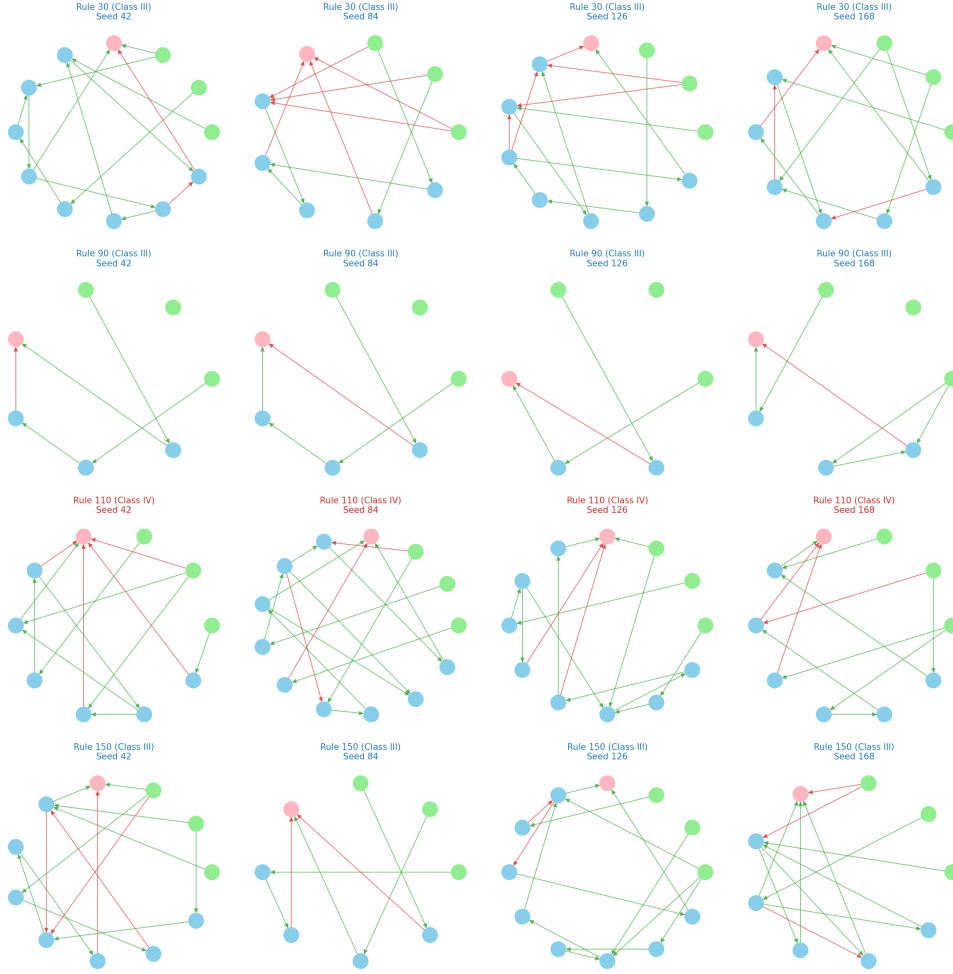


Fig. 1 NEAT-evolved networks for representative Class III and IV rules: Rule 30 (chaotic), Rule 90 (XOR), Rule 110 (Turing-complete), and Rule 150 (3-way XOR). Each network receives three inputs (left, center, right neighborhood) and produces a single output. Rule 90, implementing 2-input XOR, achieves high integration ($\Phi = 0.467$) with minimal structure, while Rule 110 requires more complex topology (6.5 hidden nodes) but exhibits near-zero integration ($\Phi = 0.007$), illustrating the inverse relationship between structural complexity and functional integration.

Rule 110 exemplifies structural complexity without functional integration

Rule 110 provides a canonical example of how universal computation can arise without functional integration. Proven to be Turing-complete [7], Rule 110 is among the most computationally powerful ECA. Yet, the networks evolved to implement this rule exhibit a distinctive combination of high structural complexity and near-zero integration.

Table 2 Functional Information profile of Rule 110. Structural values represent mean $\pm$ standard deviation across 30 evolutionary runs. I_B and I_E are analytical (rule-level) values.

Measure	Value	Type
Structural Information (I_S)	221.8 ± 221.6 bits	Network
Behavioral Information (I_B)	0.954 bits	Analytical
Effective Information (I_E)	2.046 bits	Analytical
Integration (Φ)	0.007 ± 0.038	Network
Transfer Entropy (T_E)	0.082 ± 0.021 bits	Network
Hidden nodes	6.5 ± 2.2	Network
Connections	12.3 ± 3.1	Network
DAG Depth	8.56 ± 3.3	Network

Across 30 independent evolutionary runs, networks implementing Rule 110 consistently require more structure than those implementing non-universal rules (Table 2). They contain more hidden nodes and connections and exhibit greater depth, reflecting the computational demands of universal computation. Despite this complexity, integration values remain close to zero, indicating that the learned implementations can be partitioned into relatively independent components with little loss of functional information.

A defining feature of Rule 110 is the breadth of architectures discovered by evolutionary search. Structural Information varies widely across runs, with a standard deviation comparable to the mean. This high variability indicates that many qualitatively distinct network topologies can implement the same input-output mapping, consistent with multiple valid architectural decompositions of the same computation.

Integration reflects algebraic irreducibility, not computational power

To clarify what integration measures capture, we examined representative rules across Wolfram classes (Table 3). The highest integration values are achieved not by universal rules, but by algebraically irreducible functions.

XOR-type rules exhibit the strongest integration across all ECA. These functions are irreducible in the sense that no proper subset of inputs carries information about the output, forcing tightly coupled implementations. As a result, NEAT consistently evolves architectures in which internal components are functionally interdependent, yielding high integration despite minimal structural complexity.

In contrast, rules capable of universal computation, such as Rule 110, exhibit near-zero integration despite requiring larger and deeper networks. Their learned implementations decompose into relatively independent subprocesses, indicating that computational universality does not require holistic coupling. Chaotic rules further reinforce this distinction: Rule 30 exhibits strong directional information flow, reflected in elevated transfer entropy, yet remains minimally integrated.

Table 3 Integration (Φ) and Transfer Entropy (T_E) for representative ECA rules. Values represent mean across 30 evolutionary runs.

Class	Rule	I_S (bits)	I_B (bits)	Φ	T_E (bits)
III	102 (XOR)	81	1.000	0.633	0.046
	90 (XOR)	83	1.000	0.467	0.041
	60	85	1.000	0.367	0.058
	30	214	1.000	0.000	0.097
IV	137	143	0.954	0.094	0.067
	110	222	0.954	0.007	0.082
	54	223	1.000	0.000	0.085

Modular architectures are evolutionarily accessible

Architectural variability across independent evolutionary runs provides a direct proxy for the accessibility of different computational organizations. Rules with low integration admit a wide diversity of network architectures, whereas highly integrated rules converge on constrained, repeatable solutions.

Across all 256 ECA, functional integration is negatively correlated with architectural variance. For Structural Information, this correlation is $r = -0.23$ ($p < 0.001$), with similar values observed for hidden node count and connection count. Highly integrated architectures therefore, occupy narrow regions of configuration space and can be reached only through limited evolutionary pathways. In contrast, structurally complex but low-integration rules, including Rule 110 and Rule 30, show substantial variance across runs, consistent with many accessible architectural decompositions implementing the same computation.

Functional information features recover Wolfram classes

Finally, we evaluated whether Functional Information measures capture meaningful computational structure beyond individual metrics. Hierarchical clustering using structural and weight features derived from evolved networks recovers Wolfram’s qualitative classification with high fidelity (Fig. 2).

Across all 256 rules, macro cohesion reaches 98.1%, defined as the mean fraction of rules within each Wolfram class assigned to the same cluster. Classes I, III, and IV achieve perfect cohesion, while Class II shows minimal overlap, consistent with its known heterogeneity. This recovery demonstrates that Functional Information features encode regime-level computational structure independent of explicit rule tables or spatiotemporal dynamics.

Discussion

Our results connect three ideas that are often discussed separately. These are computational universality in simple dynamical systems, information-theoretic measures of mechanism, and the evolutionary accessibility of solutions. Across all elementary

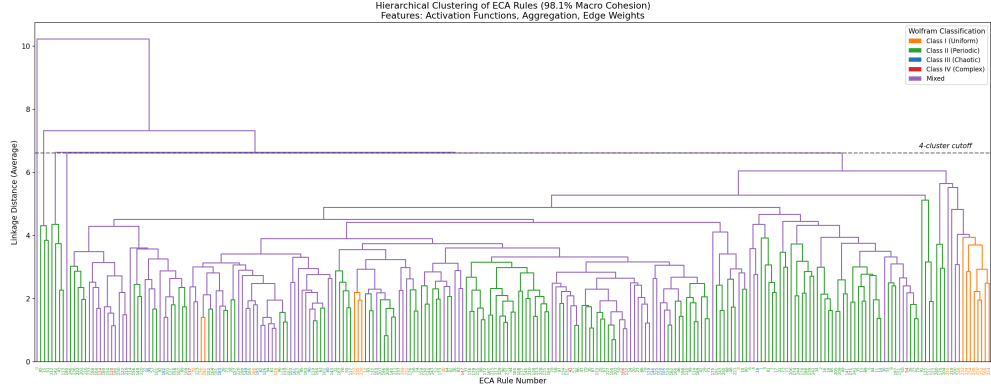


Fig. 2 Hierarchical clustering dendrogram of all 256 ECA rules based on Functional Information features. Rules are colored by Wolfram class (I=blue, II=green, III=orange, IV=red). The clustering achieves 98.1% macro cohesion, demonstrating that FI measures encode sufficient structure to recover the qualitative Wolfram classification from evolved network properties alone.

cellular automata, universal rules require substantial structural resources, yet their implementations remain weakly integrated and admit many distinct architectural realizations. By contrast, functions that are algebraically irreducible exhibit strong functional integration but occupy narrow regions of architectural space. Together, these findings suggest that functional complexity accumulates most readily when it is organized compositionally rather than through tight coupling.

Structural complexity tracks computational depth, not visual complexity

The elevated Structural Information required by Class IV rules supports the intuition that universal computation demands deeper representational capacity than trivial mappings. However, the results also clarify a long-standing ambiguity in the study of cellular automata; that is, visually complex or chaotic dynamics do not necessarily imply computational depth. Several Class III rules generate highly irregular spatiotemporal patterns, yet their local update functions are structurally modest compared with those of Class IV rules.

This distinction highlights an important separation between apparent complexity (i.e., what the system looks like when iterated) and computational complexity (i.e., what is required to implement the rule itself). In this sense, chaos is cheap to generate, whereas universality is not. Structural Information, as measured here through evolved implementations, tracks the latter rather than the former.

Integration reflects algebraic irreducibility, not universality

The strongest dissociation uncovered in this study concerns functional integration. High values of the integration measure Φ arise not in universal rules, but in algebraically irreducible ones. XOR-type rules provide the clearest example: because no

proper subset of inputs carries information about the output, any correct implementation must combine inputs jointly. NEAT responds to this constraint by evolving tightly coupled architectures, yielding high values of Φ even when the networks are small.

Universal rules achieve computational power through a different organizational strategy. Rule 110, for example, requires a deeper architecture but remains weakly integrated. Its computation decomposes into relatively independent subprocesses, allowing the system to be partitioned with little information loss. Universality, in this setting, emerges from the coordinated composition of modules rather than from holistic integration.

This distinction also sharpens the relationship between our integration measure and Integrated Information Theory (IIT). IIT treats high Φ as a signature of irreducible causal structure—information generated “from the inside.” Our results imply that such irreducibility and computational universality can be largely orthogonal. Parity functions can be highly integrated yet computationally limited, whereas universal rules can be computationally powerful while remaining decomposable at the level of mechanism.

Evolutionary accessibility favors modular organization

A central contribution of this work is to link architectural organization to evolutionary accessibility. The variance observed across independent evolutionary runs is not incidental; it reflects the structure of the underlying solution space. Highly integrated architectures converge reliably to similar forms, indicating constrained regions of configuration space. Modular, low-integration architectures admit many distinct realizations, indicating broad basins of attraction.

The observed negative correlation between integration and architectural variance provides direct evidence for this relationship. Modular architectures are accessible from many evolutionary paths: they can be discovered incrementally, modified locally, and recombined without catastrophic interference. Integrated architectures require precise coordination among components and are therefore harder to reach and harder to alter once found.

This accessibility-based perspective reframes modularity as a statistical property of search rather than a design preference. Evolution does not favor modular solutions because they are inherently superior, but because they are easier to discover and maintain under variation and selection.

Refining the law of increasing functional information

The law of increasing functional information proposes that evolving systems tend to accumulate structure that performs useful functions. Our results support this principle while refining its interpretation. Functional information does not accumulate arbitrarily; it accumulates in forms that preserve accessibility.

The inverse relationship between structural complexity and integration indicates that evolution favors a particular organization of information. High- Φ systems resist incremental change: modifying one component affects many others. Low- Φ systems

with high structural complexity decompose into modules that can be improved independently. This organizational asymmetry explains how functional complexity can increase without leading to evolutionary dead ends.

This refinement aligns with Simon’s [5] argument that nearly decomposable systems are favored because they can be assembled incrementally. In our setting, NEAT’s complexification dynamics make this process explicit: new structure is added while existing modules are preserved. Universal computation emerges not from tightly bound mechanisms, but from layered, recomposable parts.

Implications beyond cellular automata

Although ECA provides a minimal testbed, the organizational principle identified here is more general. In biological systems, from protein complexes to neural circuits, functionality is often achieved through modular composition rather than through globally integrated mechanisms. The same logic may apply to engineered systems; tasks dominated by algebraic irreducibility may benefit from tight coupling, whereas compositionally deep tasks may benefit from modular hierarchies.

From a machine learning perspective, Functional Information profiles could inform architectural bias. High-integration tasks suggest dense connectivity, while high-structure, low-integration tasks suggest depth and modular separation. This distinction offers a principled alternative to purely empirical architecture search.

Conclusion

We used evolutionary search to examine how functional information is organized in the simplest systems capable of universal computation. By evolving neural networks to implement all elementary cellular automaton rules, we showed that computational universality requires substantial structural complexity but surprisingly little functional integration. Universal rules are realized by deep yet decomposable architectures, whereas algebraically irreducible rules achieve high integration with minimal structure.

These findings refine the law of increasing functional information. Evolution does not merely accumulate more information; it accumulates information in forms that remain accessible to further adaptation. Modular architectures admit many valid implementations and are reachable from diverse evolutionary paths, while tightly integrated architectures occupy narrow regions of configuration space.

More broadly, our results separate computational power from integrative coupling. Universal computation arises not from tightly bound mechanisms, but from the composition of loosely coupled parts—a principle that may govern the organization of complex systems in biology, technology, and nature.

By reversing the usual problem—inferring the minimal computational structure from observed cellular automaton behavior—we demonstrate a principled procedure for probing discrete dynamical systems. This approach moves us closer to a microscopic understanding of how simple local rules give rise to diverse computational properties, and how complexity, modularity, and evolvability emerge from discrete rule-based systems.

Methods

Network Evolution

Neural network architectures were evolved using TensorNEAT [11], a JAX-based implementation of NeuroEvolution of Augmenting Topologies (NEAT) [8]. Separate evolutionary runs were performed for each of the 256 elementary cellular automaton (ECA) rules.

Each network receives a three-dimensional binary input vector $(s_{i-1}, s_i, s_{i+1}) \in \{0, 1\}^3$, representing the local neighborhood of a cell, and produces a single scalar output $\hat{y} \in [0, 1]$, interpreted as the predicted next state. Networks were trained to minimize binary cross-entropy loss over all eight possible input configurations.

Evolution proceeded for up to 500 generations with a population of 1,000 individuals. Sigmoid activation function was used for output nodes, while hidden nodes were permitted to use any of nine activation functions: sigmoid, tanh, ReLU, identity, sine, leaky ReLU, inverse, log, and exponential. Aggregation functions included sum and product. These choices allow expressive but compact architectures to emerge without imposing a fixed topology.

For each ECA rule, 30 independent evolutionary runs were conducted with different random seeds, yielding a distribution of network topologies implementing the same input–output mapping. Networks achieving greater than 99.9% accuracy on all eight input configurations were considered converged and included in subsequent analyses. All results reported are robust across runs; additional sensitivity analyses are provided in the Supporting Information.

Functional Information Measures

We characterize each converged network using five complementary information-theoretic measures. Let $\mathcal{N}$ denote a trained network with input set $\mathcal{X} = \{0, 1\}^3$, hidden nodes $\mathcal{H}$, and output Y .

Structural Information (I_S)

Structural Information quantifies the number of bits required to specify the network architecture, following a Minimum Description Length (MDL) principle [12]. The total description length is decomposed as

$$I_S = I_{\text{topology}} + I_{\text{weights}} + I_{\text{nodes}} \quad (1)$$

Hidden nodes contribute $I_{\text{nodes}} = |\mathcal{H}| \cdot (\log_2 9 + \log_2 2) \approx 4.17 \cdot |\mathcal{H}|$ bits, encoding activation and aggregation functions. Connections contribute $I_{\text{weights}} = |\mathcal{E}| \cdot (2 \log_2 n + 8)$ bits, encoding source, target, and an 8-bit weight value, where n is the total node count. The sparse topology itself contributes one bit per realized connection; i.e., $I_{\text{topology}} = |\mathcal{E}|$. Together, these terms quantify the descriptive complexity of the evolved implementation. A detailed breakdown is provided in the Supporting Information.

Behavioral Information (I_B)

Behavioral Information measures how much information about the input is reflected in the output. Because the networks implement deterministic mappings, this reduces to the output entropy,

$$I_B = - \sum_{y \in \{0,1\}} p(y) \log_2 p(y) \quad (2)$$

where $p(y) = \frac{1}{8} \sum_{x \in \mathcal{X}} \mathbf{1}[f(x) = y]$ is the fraction of inputs producing output y . This quantity is bounded between 0 and 1 bit. Rules producing balanced outputs (4 ones, 4 zeros) achieve maximum $I_B = 1$ bit, while constant rules (all 0s or all 1s) achieve $I_B = 0$. Notably, I_B is a property of the ECA rule itself, not the implementing network, and can be computed directly from the rule’s truth table.

Effective Information (I_E)

Effective Information quantifies causal specificity under intervention. Following prior work [9, 13], we compute EI by intervening uniformly over the input space and measuring the uncertainty of the resulting output. In this setting—deterministic, feedforward networks with uniform interventions—EI simplifies to

$$I_E = \log_2 |\mathcal{X}| - H(Y) = 3 - H(Y) \quad (3)$$

Thus, EI reflects output degeneracy: rules for which many inputs map to the same output exhibit lower EI. While EI collapses to a rule-level quantity here, this simplification arises from the restricted setting and does not hold in general.

Integration (Φ)

To quantify information integration, we use an information-loss-based measure inspired by Integrated Information Theory [10, 14]. Integration is defined as the minimum loss of mutual information between inputs and outputs induced by partitioning the network’s hidden nodes into two non-overlapping subsets and severing connections between them.

For each bipartition π of $\mathcal{H}$, we compute the information loss Δ_π and define

$$\Phi = \frac{\min_\pi \Delta_\pi}{I(X; Y)}. \quad (4)$$

This normalized measure lies in $[0, 1]$, enabling comparison across rules with different output entropies. For the small networks produced by NEAT (typically fewer than 10 hidden nodes), all bipartitions can be enumerated exactly. This formulation captures whether a computation can be decomposed into independent subcomputations, rather than implementing the full causal repertoire formalism of IIT.

Transfer Entropy (T_E)

Transfer Entropy [15] is used here as a directed information-flow measure between network components, rather than a temporal memory measure. For a network processing input X through hidden layer H to output Y :

$$T_{X \rightarrow H} = I(H; X | H^-) \quad (5)$$

$$T_{H \rightarrow Y} = I(Y; H | Y^-) \quad (6)$$

where H^- and Y^- denote previous states. For feedforward networks (which NEAT typically produces for this task), this simplifies to:

$$T_E = I(X; H) + I(H; Y | X) \quad (7)$$

This measures how much information flows through the hidden layer versus being transmitted directly from input to output.

For networks with skip connections (direct input-to-output edges), separate quantification is made of mediated information (passing through hidden nodes) and direct information (bypassing the hidden layer).

Wolfram Class Assignments

We adopt the standard Wolfram classification of elementary cellular automata. All 256 rules are analyzed individually, with aggregates computed per class. Although the rules reduce to 88 equivalence classes under symmetry, treating them separately preserves rule-level variation.

Statistical Analysis

For each rule, the mean and standard deviation of each Functional Information (FI) measures were computed across the 30 evolutionary runs. Class-level differences were compared using Kruskal-Wallis tests, with pairwise comparisons performed using Mann-Whitney U tests, and effect sizes reported as Cohen’s d . Correlations between measures were assessed using Pearson’s r .

Hierarchical clustering was applied to multidimensional FI feature vectors to evaluate whether FI alone recovers Wolfram class structure. Robustness analyses, including feature ablations and alternative clustering methods, are reported in the Supporting Information.

Acknowledgements.

Declarations

- **Conflict of interest/Competing interests:** The authors declare no competing interests.
- **Ethics approval:** Not applicable.
- **Consent for publication:** Not applicable.
- **Data availability:** Data and code will be made available upon acceptance.
- **Code availability:** Code will be made available upon acceptance.
- **Author contribution:** J.M.A.M. designed and performed research; J.M.A.M., E.F.T.L., and C.P.M. wrote the manuscript. All authors approved the final version.

References

- [1] Michael L. Wong *et al.* On the Roles of Function and Selection in Evolving Systems. *Proceedings of the National Academy of Sciences* **120**, e2310223120 (2023).
- [2] Chen, X., Zaro, J. & Shen, W.-C. Fusion protein linkers: Property, design and functionality. *Adv Drug Deliv Rev.* **65** (2014).
- [3] Aguirre, M., Spence, J. P., Sella, G. & Pritchard, J. K. Gene regulatory network structure informs the distribution of perturbation effects. *PLoS Comput Biol.* **21** (2025).
- [4] Luo, L. Architectures of neuronal circuits. *Science* **373** (2021).
- [5] Herbert A. Simon. The Architecture of Complexity. *Proceedings of the American Philosophical Society* **106**, 467–482 (1962). URL <https://www.jstor.org/stable/985254>.
- [6] Stephen Wolfram. *A New Kind of Science* (Wolfram Media, Champaign, IL, 2002).
- [7] Matthew Cook. Universality in Elementary Cellular Automata. *Complex Systems* **15**, 1–40 (2004).
- [8] Kenneth O. Stanley & Risto Miikkulainen. Evolving Neural Networks through Augmenting Topologies. *Evolutionary Computation* **10**, 99–127 (2002).
- [9] Erik P. Hoel, Larissa Albantakis & Giulio Tononi. Quantifying Causal Emergence Shows That Macro Can Beat Micro. *Proceedings of the National Academy of Sciences* **110**, 19790–19795 (2013).
- [10] Giulio Tononi. Consciousness as Integrated Information: A Provisional Manifesto. *The Biological Bulletin* **215**, 216–242 (2008).
- [11] Wang, C., Wang, Y. & Cheng, R. Tensorized NeuroEvolution of augmenting topologies for GPU acceleration. *Proc. Genet. Evol. Comput. Conf. (GECCO), Best Paper Award* 1–9 (2024).
- [12] Jorma Rissanen. Modeling by Shortest Data Description. *Automatica* **14**, 465–471 (1978).
- [13] Giulio Tononi, Olaf Sporns & Gerald M. Edelman. Measures of Degeneracy and Redundancy in Biological Networks. *Proceedings of the National Academy of Sciences* **96**, 3257–3262 (1999).
- [14] Masafumi Oizumi, Larissa Albantakis & Giulio Tononi. From the Phenomenology to the Mechanisms of Consciousness: Integrated Information Theory 3.0. *PLOS*

Computational Biology **10**, e1003588 (2014).

- [15] Thomas Schreiber. Measuring Information Transfer. *Physical Review Letters* **85**, 461–464 (2000).

Structural Information Formula

Table S1 provides the detailed breakdown of the Structural Information (I_S) formula, demonstrating the information-theoretic justification for each component. The formula follows the principle that specifying a choice among N options requires $\log_2 N$ bits.

Table S1 Structural Information (I_S) formula breakdown with Rule 110 example. Rule 110 has $|\mathcal{H}| = 6.5$ hidden nodes, $|\mathcal{E}| = 12.3$ connections, and $n = 10.5$ total nodes on average.

Component	Formula	Justification	Example (Rule 110)
<i>Per-node encoding</i>			
Activation	$\log_2 9 \approx 3.17$ bits	9 activation functions	
Aggregation	$\log_2 2 = 1$ bit	2 aggregation functions	
Total per node	≈ 4.17 bits		
<i>Per-connection encoding</i>			
Source node	$\log_2 n$ bits	n possible sources	$\log_2 10.5 \approx 3.39$
Target node	$\log_2 n$ bits	n possible targets	$\log_2 10.5 \approx 3.39$
Weight value	8 bits	256 weight levels	8.00
Total per conn	$2 \log_2 n + 8$ bits		≈ 14.78
<i>Total I_S</i>			
I_{nodes}	$ \mathcal{H} \times 4.17$	Hidden nodes only	$6.5 \times 4.17 = 27.1$
I_{weights}	$ \mathcal{E} \times (2 \log_2 n + 8)$	All connections	$12.3 \times 14.78 = 182.3$
I_{topology}	$ \mathcal{E} $	Sparse encoding	12.3
Total I_S			221.8 bits

The 8-bit weight precision (256 levels) is sufficient for neural network weights, as finer distinctions typically do not affect network behavior. This principled approach yields I_S values that scale appropriately with network complexity: simple rules like Rule 0 require ~ 69 bits, while complex Class IV rules require ~ 208 – 262 bits.

S1 Complete Functional Information Data

Tables S2–S5 present the complete Functional Information profile for all 256 ECA rules. Class designations follow the standard Wolfram classification. I_S values represent mean Structural Information across 30 evolutionary runs. I_B and I_E are analytical values computed from the rule’s truth table. Φ and T_E are mean values from evolved networks.

S2 Analytical Functional Information

For completeness, Table S6 shows the analytical Behavioral Information (I_B) and Effective Information (I_E) for rules grouped by output balance. These values are computed directly from the ECA truth tables.

Table S2 Functional Information for ECA rules 0–63.

R	Cl	I_S	I_B	I_E	Φ	T_E	R	Cl	I_S	I_B	I_E	Φ	T_E
0	I	69	0.00	3.00	0.00	0.03	32	I	178	0.54	2.46	0.01	0.06
1	II	167	0.54	2.46	0.00	0.09	33	II	233	0.81	2.19	0.01	0.09
2	II	166	0.54	2.46	0.03	0.07	34	II	117	0.81	2.19	0.03	0.08
3	II	131	0.81	2.19	0.02	0.08	35	II	209	0.95	2.05	0.00	0.08
4	II	163	0.54	2.46	0.03	0.08	36	II	221	0.81	2.19	0.02	0.09
5	II	127	0.81	2.19	0.05	0.09	37	II	242	0.95	2.05	0.00	0.08
6	II	193	0.81	2.19	0.00	0.08	38	II	195	0.95	2.05	0.01	0.09
7	II	200	0.95	2.05	0.00	0.10	39	II	216	1.00	2.00	0.00	0.09
8	I	177	0.54	2.46	0.02	0.07	40	I	235	0.81	2.19	0.02	0.08
9	II	223	0.81	2.19	0.04	0.09	41	IV	262	0.95	2.05	0.00	0.09
10	II	115	0.81	2.19	0.07	0.07	42	II	155	0.95	2.05	0.02	0.10
11	II	195	0.95	2.05	0.00	0.08	43	II	214	1.00	2.00	0.01	0.09
12	II	113	0.81	2.19	0.08	0.08	44	II	202	0.95	2.05	0.00	0.09
13	II	189	0.95	2.05	0.01	0.08	45	III	214	1.00	2.00	0.02	0.08
14	II	204	0.95	2.05	0.01	0.09	46	II	221	1.00	2.00	0.00	0.11
15	II	70	1.00	2.00	0.00	0.06	47	II	191	0.95	2.05	0.00	0.10
16	II	171	0.54	2.46	0.02	0.08	48	II	119	0.81	2.19	0.04	0.07
17	II	128	0.81	2.19	0.04	0.09	49	II	206	0.95	2.05	0.01	0.08
18	III	201	0.81	2.19	0.00	0.09	50	II	183	0.95	2.05	0.01	0.09
19	II	213	0.95	2.05	0.00	0.09	51	II	60	1.00	2.00	0.00	0.04
20	II	205	0.81	2.19	0.01	0.08	52	II	205	0.95	2.05	0.00	0.09
21	II	214	0.95	2.05	0.00	0.09	53	II	228	1.00	2.00	0.00	0.10
22	III	242	0.95	2.05	0.00	0.09	54	IV	223	1.00	2.00	0.00	0.09
23	II	229	1.00	2.00	0.00	0.08	55	II	186	0.95	2.05	0.03	0.08
24	II	216	0.81	2.19	0.02	0.09	56	II	198	0.95	2.05	0.01	0.10
25	II	201	0.95	2.05	0.00	0.10	57	II	242	1.00	2.00	0.00	0.08
26	II	214	0.95	2.05	0.01	0.09	58	II	224	1.00	2.00	0.00	0.11
27	II	197	1.00	2.00	0.00	0.11	59	II	195	0.95	2.05	0.00	0.07
28	II	205	0.95	2.05	0.01	0.09	60	III	85	1.00	2.00	0.37	0.06
29	II	209	1.00	2.00	0.00	0.10	61	II	158	0.95	2.05	0.09	0.07
30	III	214	1.00	2.00	0.00	0.10	62	II	156	0.95	2.05	0.10	0.08
31	II	180	0.95	2.05	0.04	0.08	63	II	109	0.81	2.19	0.02	0.01

S3 Structural Features of Evolved Networks

Tables S7–S10 present the structural features extracted from NEAT-evolved networks for all 256 ECA rules. Values represent means across 30 evolutionary runs. H = hidden nodes, E = edges, D = depth, In = inputs used, A = activation functions, P = product aggregation fraction.

S4 Clustering Analysis Details

S4.1 Feature Groups

Table S11 defines the feature groups used in clustering analysis. Information-theoretic features derive from network complexity analysis; graph structure features derive from NEAT network topology.

Table S3 Functional Information for ECA rules 64–127.

R	Cl	I_S	I_B	I_E	Φ	T_E	R	Cl	I_S	I_B	I_E	Φ	T_E
64	I	175	0.54	2.46	0.01	0.08	96	I	208	0.81	2.19	0.01	0.08
65	II	218	0.81	2.19	0.01	0.08	97	II	256	0.95	2.05	0.00	0.09
66	II	211	0.81	2.19	0.06	0.09	98	II	207	0.95	2.05	0.04	0.10
67	II	213	0.95	2.05	0.00	0.09	99	II	209	1.00	2.00	0.00	0.10
68	II	113	0.81	2.19	0.05	0.08	100	II	193	0.95	2.05	0.00	0.09
69	II	196	0.95	2.05	0.01	0.09	101	III	219	1.00	2.00	0.00	0.09
70	II	199	0.95	2.05	0.01	0.09	102	III	81	1.00	2.00	0.63	0.05
71	II	200	1.00	2.00	0.01	0.10	103	II	161	0.95	2.05	0.09	0.07
72	II	222	0.81	2.19	0.01	0.08	104	II	238	0.95	2.05	0.00	0.08
73	III	230	0.95	2.05	0.00	0.09	105	III	170	1.00	2.00	0.20	0.08
74	II	192	0.95	2.05	0.00	0.08	106	III	196	1.00	2.00	0.03	0.07
75	III	215	1.00	2.00	0.00	0.09	107	II	249	0.95	2.05	0.00	0.08
76	II	187	0.95	2.05	0.02	0.09	108	II	203	1.00	2.00	0.01	0.08
77	II	206	1.00	2.00	0.00	0.10	109	III	244	0.95	2.05	0.00	0.07
78	II	220	1.00	2.00	0.00	0.09	110	IV	222	0.95	2.05	0.01	0.08
79	II	177	0.95	2.05	0.01	0.10	111	II	231	0.81	2.19	0.01	0.04
80	II	113	0.81	2.19	0.07	0.08	112	II	160	0.95	2.05	0.03	0.08
81	II	205	0.95	2.05	0.00	0.10	113	II	206	1.00	2.00	0.00	0.09
82	II	194	0.95	2.05	0.02	0.08	114	II	208	1.00	2.00	0.00	0.10
83	II	210	1.00	2.00	0.02	0.10	115	II	201	0.95	2.05	0.00	0.09
84	II	207	0.95	2.05	0.01	0.09	116	II	203	1.00	2.00	0.00	0.09
85	II	72	1.00	2.00	0.00	0.06	117	II	182	0.95	2.05	0.01	0.10
86	III	202	1.00	2.00	0.00	0.10	118	II	161	0.95	2.05	0.08	0.08
87	II	182	0.95	2.05	0.03	0.10	119	II	110	0.81	2.19	0.03	0.02
88	II	207	0.95	2.05	0.00	0.08	120	III	211	1.00	2.00	0.03	0.07
89	III	218	1.00	2.00	0.00	0.08	121	II	251	0.95	2.05	0.02	0.08
90	III	83	1.00	2.00	0.47	0.04	122	III	197	0.95	2.05	0.01	0.08
91	II	168	0.95	2.05	0.09	0.06	123	II	228	0.81	2.19	0.02	0.05
92	II	232	1.00	2.00	0.01	0.10	124	IV	209	0.95	2.05	0.00	0.09
93	II	191	0.95	2.05	0.00	0.09	125	II	235	0.81	2.19	0.01	0.04
94	II	169	0.95	2.05	0.13	0.08	126	III	205	0.81	2.19	0.01	0.05
95	II	104	0.81	2.19	0.04	0.02	127	II	184	0.54	2.46	0.00	0.02

S4.2 Top Clustering Results

Table S12 presents the top 10 clustering results by macro cohesion using information-theoretic features only.

S4.3 Feature Importance

Table S13 shows the frequency of each feature group in top-50 clustering results, indicating discriminative power for Wolfram classification. Figure S1 visualizes this analysis.

S4.4 XOR Rules and Integration

Table S14 shows the relationship between XOR-type rules and integration. XOR functions are algebraically irreducible—neither input alone determines the output—forcing NEAT to evolve tightly coupled networks with high Φ .

Table S4 Functional Information for ECA rules 128–191.

R	Cl	I_S	I_B	I_E	Φ	T_E	R	Cl	I_S	I_B	I_E	Φ	T_E
128	I	182	0.54	2.46	0.00	0.07	160	I	117	0.81	2.19	0.03	0.06
129	III	209	0.81	2.19	0.00	0.09	161	III	145	0.95	2.05	0.19	0.10
130	II	227	0.81	2.19	0.04	0.07	162	II	178	0.95	2.05	0.03	0.10
131	II	207	0.95	2.05	0.00	0.09	163	II	223	1.00	2.00	0.00	0.09
132	II	226	0.81	2.19	0.01	0.09	164	II	157	0.95	2.05	0.11	0.09
133	II	216	0.95	2.05	0.01	0.08	165	III	126	1.00	2.00	0.20	0.08
134	II	223	0.95	2.05	0.00	0.08	166	II	228	1.00	2.00	0.02	0.10
135	III	219	1.00	2.00	0.02	0.08	167	II	194	0.95	2.05	0.01	0.08
136	I	104	0.81	2.19	0.05	0.06	168	I	187	0.95	2.05	0.03	0.09
137	IV	143	0.95	2.05	0.09	0.07	169	III	208	1.00	2.00	0.01	0.09
138	II	187	0.95	2.05	0.01	0.10	170	II	64	1.00	2.00	0.00	0.04
139	II	225	1.00	2.00	0.00	0.10	171	II	181	0.95	2.05	0.00	0.09
140	II	193	0.95	2.05	0.02	0.09	172	II	203	1.00	2.00	0.02	0.09
141	II	241	1.00	2.00	0.00	0.10	173	II	215	0.95	2.05	0.01	0.09
142	II	221	1.00	2.00	0.00	0.10	174	II	190	0.95	2.05	0.01	0.08
143	II	152	0.95	2.05	0.05	0.08	175	II	116	0.81	2.19	0.05	0.05
144	II	225	0.81	2.19	0.00	0.09	176	II	180	0.95	2.05	0.01	0.09
145	III	209	0.95	2.05	0.00	0.09	177	II	219	1.00	2.00	0.01	0.09
146	III	226	0.95	2.05	0.00	0.09	178	II	201	1.00	2.00	0.00	0.10
147	IV	192	1.00	2.00	0.00	0.09	179	II	172	0.95	2.05	0.01	0.09
148	II	221	0.95	2.05	0.02	0.09	180	II	251	1.00	2.00	0.00	0.08
149	III	216	1.00	2.00	0.00	0.07	181	II	187	0.95	2.05	0.01	0.08
150	III	192	1.00	2.00	0.07	0.07	182	III	240	0.95	2.05	0.00	0.08
151	II	227	0.95	2.05	0.02	0.08	183	III	209	0.81	2.19	0.00	0.04
152	II	171	0.95	2.05	0.04	0.08	184	II	193	1.00	2.00	0.02	0.09
153	III	128	1.00	2.00	0.03	0.06	185	II	222	0.95	2.05	0.00	0.09
154	II	210	1.00	2.00	0.00	0.08	186	II	188	0.95	2.05	0.00	0.10
155	II	224	0.95	2.05	0.00	0.10	187	II	117	0.81	2.19	0.06	0.04
156	II	199	1.00	2.00	0.02	0.08	188	II	214	0.95	2.05	0.01	0.08
157	II	192	0.95	2.05	0.01	0.10	189	II	236	0.81	2.19	0.00	0.07
158	II	240	0.95	2.05	0.00	0.08	190	II	237	0.81	2.19	0.00	0.07
159	II	246	0.81	2.19	0.01	0.05	191	II	172	0.54	2.46	0.02	0.03

S5 Variance-Correlation Analysis

The variance in network architectures across evolutionary runs provides evidence for the modularity-evolvability hypothesis. Table S15 shows the correlations between Integration (Φ) and architectural variance measures.

Table S16 shows variance statistics grouped by Wolfram class. Class IV rules exhibit the highest variance, consistent with their modular structure admitting multiple valid decompositions.

Table S17 shows variance for individual key rules, illustrating the contrast between XOR-type rules (high Φ , low variance) and Class IV rules (low Φ , high variance).

Table S5 Functional Information for ECA rules 192–255.

R	Cl	I_S	I_B	I_E	Φ	T_E	R	Cl	I_S	I_B	I_E	Φ	T_E
192	I	96	0.81	2.19	0.02	0.06	224	I	176	0.95	2.05	0.01	0.09
193	II	149	0.95	2.05	0.20	0.07	225	III	213	1.00	2.00	0.00	0.11
194	II	150	0.95	2.05	0.06	0.08	226	II	214	1.00	2.00	0.00	0.09
195	III	131	1.00	2.00	0.07	0.08	227	II	222	0.95	2.05	0.00	0.09
196	II	181	0.95	2.05	0.00	0.08	228	II	207	1.00	2.00	0.00	0.10
197	II	235	1.00	2.00	0.00	0.10	229	II	212	0.95	2.05	0.01	0.08
198	II	210	1.00	2.00	0.00	0.11	230	II	211	0.95	2.05	0.01	0.08
199	II	193	0.95	2.05	0.02	0.08	231	II	203	0.81	2.19	0.02	0.06
200	II	193	0.95	2.05	0.03	0.08	232	II	230	1.00	2.00	0.00	0.11
201	II	233	1.00	2.00	0.00	0.09	233	II	237	0.95	2.05	0.00	0.07
202	II	221	1.00	2.00	0.01	0.10	234	I	216	0.95	2.05	0.00	0.10
203	II	204	0.95	2.05	0.02	0.09	235	I	213	0.81	2.19	0.00	0.07
204	II	76	1.00	2.00	0.00	0.05	236	II	207	0.95	2.05	0.01	0.09
205	II	198	0.95	2.05	0.00	0.08	237	II	220	0.81	2.19	0.00	0.08
206	II	181	0.95	2.05	0.01	0.10	238	I	124	0.81	2.19	0.02	0.04
207	II	127	0.81	2.19	0.03	0.04	239	I	169	0.54	2.46	0.02	0.04
208	II	166	0.95	2.05	0.01	0.09	240	II	63	1.00	2.00	0.00	0.04
209	II	213	1.00	2.00	0.00	0.09	241	II	199	0.95	2.05	0.00	0.09
210	II	212	1.00	2.00	0.02	0.07	242	II	210	0.95	2.05	0.00	0.08
211	II	190	0.95	2.05	0.02	0.08	243	II	120	0.81	2.19	0.04	0.04
212	II	222	1.00	2.00	0.00	0.08	244	II	192	0.95	2.05	0.00	0.09
213	II	171	0.95	2.05	0.01	0.09	245	II	120	0.81	2.19	0.03	0.05
214	II	224	0.95	2.05	0.00	0.08	246	II	225	0.81	2.19	0.00	0.07
215	II	215	0.81	2.19	0.02	0.05	247	II	193	0.54	2.46	0.00	0.02
216	II	227	1.00	2.00	0.01	0.10	248	I	196	0.95	2.05	0.01	0.09
217	II	218	0.95	2.05	0.01	0.09	249	I	217	0.81	2.19	0.01	0.08
218	II	219	0.95	2.05	0.01	0.08	250	I	118	0.81	2.19	0.03	0.05
219	II	203	0.81	2.19	0.00	0.08	251	I	176	0.54	2.46	0.04	0.04
220	II	194	0.95	2.05	0.00	0.09	252	I	127	0.81	2.19	0.00	0.05
221	II	119	0.81	2.19	0.09	0.02	253	I	175	0.54	2.46	0.02	0.05
222	II	221	0.81	2.19	0.00	0.06	254	I	179	0.54	2.46	0.02	0.07
223	II	182	0.54	2.46	0.02	0.03	255	I	74	0.00	3.00	0.00	0.02

Table S6 Analytical Functional Information by output balance. $I_B = H(Y)$ is the output entropy; $I_E = 3 - H(Y)$ is the effective information.

Ones	Zeros	I_B (bits)	I_E (bits)	Rule Count
0	8	0.000	3.000	1
1	7	0.544	2.456	8
2	6	0.811	2.189	28
3	5	0.954	2.046	56
4	4	1.000	2.000	70
5	3	0.954	2.046	56
6	2	0.811	2.189	28
7	1	0.544	2.456	8
8	0	0.000	3.000	1

Table S7 Structural features for ECA rules 0–63.

R	Cl	H	E	D	In	A	P	R	Cl	H	E	D	In	A	P
0	I	2.0	4	2.0	2.2	3.8	1.0	32	I	5.0	10	3.8	3.0	5.0	1.0
1	II	3.9	10	3.6	3.0	4.9	0.9	33	II	6.5	13	6.4	3.0	5.7	1.0
2	II	4.4	10	4.2	3.0	5.5	1.0	34	II	3.5	7	4.0	2.3	4.9	1.0
3	II	3.5	8	3.8	2.5	4.8	0.9	35	II	5.7	12	6.7	3.0	5.2	1.0
4	II	4.3	10	4.6	3.0	5.7	0.9	36	II	6.4	12	5.5	3.0	5.5	1.0
5	II	3.4	8	3.5	2.4	4.6	1.0	37	II	7.0	13	6.3	3.0	5.6	1.0
6	II	4.9	11	4.2	3.0	5.3	1.0	38	II	5.7	11	6.0	3.0	5.5	1.0
7	II	5.5	11	6.0	3.0	5.1	0.9	39	II	5.9	12	5.8	3.0	5.5	1.0
8	I	5.2	10	4.0	3.0	5.1	0.9	40	I	7.0	13	5.7	3.0	5.7	1.0
9	II	6.1	13	5.7	3.0	5.5	1.0	41	IV	7.7	14	8.6	3.0	5.9	1.0
10	II	3.3	7	4.2	2.4	5.1	0.9	42	II	4.3	9	4.6	3.0	5.0	1.0
11	II	5.4	11	6.6	3.0	5.0	0.9	43	II	6.0	12	8.0	3.0	4.8	1.0
12	II	3.1	7	3.3	2.3	4.8	1.0	44	II	5.7	12	5.9	3.0	5.5	1.0
13	II	5.1	11	6.1	3.0	5.0	1.0	45	III	5.8	12	5.9	3.0	5.7	1.0
14	II	5.3	12	7.2	3.0	5.1	1.0	46	II	6.1	13	7.8	3.0	5.3	1.0
15	II	1.8	5	1.3	2.3	4.2	0.9	47	II	5.6	11	6.6	3.0	5.3	1.0
16	II	4.5	10	4.5	3.0	5.3	1.0	48	II	3.5	7	3.7	2.4	4.9	0.9
17	II	3.7	8	3.8	2.5	4.8	0.9	49	II	5.3	12	6.0	3.0	5.1	0.9
18	III	5.3	12	5.2	3.0	5.6	1.0	50	II	4.5	11	5.4	3.0	5.1	1.0
19	II	5.8	12	6.3	3.0	5.3	1.0	51	II	1.6	4	0.7	2.2	4.2	0.9
20	II	5.4	12	4.9	3.0	5.6	1.0	52	II	6.2	11	5.8	3.0	5.7	1.0
21	II	5.8	12	5.7	3.0	5.4	1.0	53	II	6.6	13	6.6	3.0	5.7	1.0
22	III	6.9	13	6.2	3.0	5.6	1.0	54	IV	7.0	12	8.2	3.0	5.3	1.0
23	II	6.3	13	6.5	3.0	5.2	1.0	55	II	5.4	11	4.1	3.0	5.2	1.0
24	II	5.7	12	5.4	3.0	5.6	1.0	56	II	5.8	11	6.2	3.0	5.4	0.9
25	II	6.0	11	5.4	3.0	5.8	1.0	57	II	7.0	13	7.1	3.0	6.1	1.0
26	II	6.3	12	5.7	3.0	5.9	1.0	58	II	6.4	13	9.1	3.0	5.8	1.0
27	II	5.3	11	5.2	3.0	5.5	0.9	59	II	5.5	11	6.5	3.0	5.6	1.0
28	II	5.9	12	6.2	3.0	5.3	1.0	60	III	2.8	5	3.0	2.0	4.7	1.0
29	II	5.9	12	5.2	3.0	5.4	1.0	61	II	4.6	9	5.0	3.0	5.1	1.0
30	III	6.1	12	6.4	3.0	5.6	1.0	62	II	4.3	9	4.9	3.0	5.3	1.0
31	II	5.0	10	4.0	3.0	5.1	1.0	63	II	3.2	7	4.1	2.1	4.6	1.0

Table S8 Structural features for ECA rules 64–127.

R	Cl	H	E	D	In	A	P	R	Cl	H	E	D	In	A	P
64	I	4.8	10	4.5	3.0	5.3	1.0	96	I	6.4	12	5.3	3.0	5.4	1.0
65	II	5.9	12	5.8	3.0	5.3	1.0	97	II	7.2	14	8.1	3.0	5.7	1.0
66	II	5.8	12	6.5	3.0	5.4	1.0	98	II	6.3	12	6.0	3.0	5.4	1.0
67	II	6.1	12	5.5	3.0	5.6	1.0	99	II	5.9	12	6.7	3.0	5.2	1.0
68	II	3.1	7	3.6	2.4	4.8	0.9	100	II	5.5	11	5.7	3.0	5.7	1.0
69	II	5.4	11	6.7	3.0	5.1	1.0	101	III	6.5	12	7.0	3.0	6.1	1.0
70	II	5.8	11	6.2	3.0	5.6	1.0	102	III	2.5	5	2.9	2.1	4.2	1.0
71	II	5.8	11	5.2	3.0	5.4	1.0	103	II	4.7	9	5.0	3.0	4.7	1.0
72	II	6.4	12	5.3	3.0	5.6	1.0	104	II	6.8	13	7.4	3.0	5.4	1.0
73	III	6.5	13	7.9	3.0	5.3	0.9	105	III	5.3	10	5.0	3.0	4.9	1.0
74	II	5.5	11	5.6	3.0	5.6	1.0	106	III	5.4	11	6.6	3.0	5.5	1.0
75	III	6.3	12	6.9	3.0	5.4	1.0	107	II	6.9	14	7.8	3.0	5.8	1.0
76	II	5.2	11	5.3	3.0	5.1	1.0	108	II	5.5	12	5.8	3.0	5.3	1.0
77	II	5.4	12	7.3	3.0	5.0	0.9	109	III	6.8	14	7.9	3.0	5.9	1.0
78	II	6.5	12	7.4	3.0	5.5	1.0	110	IV	6.5	12	8.6	3.0	5.9	1.0
79	II	4.9	10	5.4	3.0	5.3	1.0	111	II	6.8	13	5.5	3.0	5.9	1.0
80	II	3.2	7	3.6	2.5	4.8	0.9	112	II	4.4	9	5.2	3.0	5.3	1.0
81	II	5.2	12	6.1	3.0	5.1	0.9	113	II	5.3	12	7.4	3.0	4.8	1.0
82	II	5.7	11	5.0	3.0	5.4	1.0	114	II	5.5	12	9.0	3.0	5.3	1.0
83	II	6.1	12	5.6	3.0	5.5	1.0	115	II	6.0	11	6.8	3.0	5.3	1.0
84	II	5.4	12	6.6	3.0	4.9	0.9	116	II	5.5	12	7.8	3.0	5.4	1.0
85	II	1.7	5	1.2	2.5	4.3	0.9	117	II	5.2	10	6.0	3.0	5.3	1.0
86	III	5.8	11	6.8	3.0	5.6	1.0	118	II	4.6	9	4.8	3.0	5.1	1.0
87	II	4.8	11	3.6	3.0	5.5	1.0	119	II	3.3	7	4.0	2.0	4.6	1.0
88	II	5.9	12	6.5	3.0	5.5	1.0	120	III	6.1	12	6.9	3.0	5.3	1.0
89	III	6.0	12	7.0	3.0	5.8	1.0	121	II	7.0	14	8.4	3.0	5.7	1.0
90	III	2.6	5	3.1	2.2	4.4	1.0	122	III	5.7	11	7.4	3.0	5.7	1.0
91	II	4.8	10	5.4	3.0	5.3	1.0	123	II	6.7	13	5.3	3.0	5.9	1.0
92	II	6.5	13	9.0	3.0	6.0	1.0	124	IV	6.2	12	8.2	3.0	5.8	1.0
93	II	5.5	11	6.1	3.0	5.6	1.0	125	II	6.9	13	5.5	3.0	6.1	1.0
94	II	5.0	10	5.7	3.0	5.2	1.0	126	III	5.8	12	5.2	3.0	5.4	1.0
95	II	3.0	6	4.1	2.2	4.4	1.0	127	II	5.0	11	4.8	3.0	5.2	1.0

Table S9 Structural features for ECA rules 128–191.

R	Cl	H	E	D	In	A	P	R	Cl	H	E	D	In	A	P
128	I	5.1	11	5.3	3.0	5.2	1.0	160	I	3.4	7	3.9	2.2	4.9	1.0
129	III	5.6	12	5.0	3.0	5.6	1.0	161	III	4.1	9	4.5	3.0	4.9	1.0
130	II	6.6	13	5.8	3.0	5.8	1.0	162	II	5.1	10	6.1	3.0	5.1	1.0
131	II	6.3	12	6.4	3.0	5.8	1.0	163	II	6.6	12	8.9	3.0	5.1	1.0
132	II	6.6	13	5.4	3.0	5.7	1.0	164	II	4.6	9	4.5	3.0	5.1	1.0
133	II	6.6	12	6.9	3.0	5.5	1.0	165	III	3.8	7	4.8	2.4	4.9	1.0
134	II	6.3	13	7.6	3.0	5.5	1.0	166	II	6.3	13	9.3	3.0	5.2	1.0
135	III	6.2	12	6.7	3.0	5.4	1.0	167	II	5.5	11	6.2	3.0	5.4	1.0
136	I	3.0	6	3.4	2.2	4.8	1.0	168	I	5.3	11	7.1	3.0	5.1	0.9
137	IV	3.9	9	4.3	3.0	4.7	1.0	169	III	5.6	12	9.6	3.0	5.2	1.0
138	II	5.5	11	6.4	3.0	5.3	1.0	170	II	1.7	4	3.4	2.1	4.3	0.9
139	II	6.0	13	9.7	3.0	4.9	0.9	171	II	4.7	11	9.0	3.0	4.8	1.0
140	II	5.4	11	6.0	3.0	5.2	1.0	172	II	5.7	12	7.2	3.0	5.6	1.0
141	II	6.6	14	8.7	3.0	5.3	1.0	173	II	6.0	12	6.6	3.0	5.7	1.0
142	II	6.2	13	9.0	3.0	4.5	1.0	174	II	5.1	11	9.5	3.0	5.0	1.0
143	II	4.2	9	4.7	3.0	4.8	1.0	175	II	3.5	7	5.6	2.4	4.7	0.9
144	II	6.7	13	5.6	3.0	5.9	1.0	176	II	4.9	10	5.9	3.0	5.3	1.0
145	II	6.2	12	5.6	3.0	5.4	1.0	177	II	5.9	13	7.2	3.0	5.6	1.0
146	III	6.4	13	7.5	3.0	5.5	1.0	178	II	5.1	12	8.1	3.0	4.7	1.0
147	IV	5.7	11	5.5	3.0	5.7	1.0	179	II	4.9	10	6.0	3.0	5.5	1.0
148	II	6.4	12	8.5	3.0	5.6	1.0	180	II	7.1	14	9.2	3.0	6.0	1.0
149	III	6.0	12	6.1	3.0	5.3	1.0	181	II	5.3	11	5.7	3.0	5.8	1.0
150	III	5.7	11	6.4	3.0	5.1	1.0	182	III	6.4	14	8.6	3.0	5.6	1.0
151	II	6.6	13	5.0	3.0	5.0	1.0	183	III	6.0	12	4.8	3.0	5.6	1.0
152	II	4.9	10	4.5	3.0	5.1	1.0	184	II	5.5	11	6.7	3.0	5.4	1.0
153	III	3.7	8	4.9	2.5	5.0	1.0	185	II	6.6	12	7.1	3.0	5.6	0.9
154	II	5.9	12	8.5	3.0	5.8	1.0	186	II	4.7	11	9.6	3.0	4.8	1.0
155	II	6.7	12	7.2	3.0	5.8	1.0	187	II	3.3	7	5.8	2.6	4.6	1.0
156	II	5.8	11	8.9	3.0	5.8	0.9	188	II	6.1	12	9.1	3.0	6.0	1.0
157	II	5.5	11	7.0	3.0	5.1	1.0	189	II	6.6	13	7.8	3.0	6.2	1.0
158	II	6.7	13	8.1	3.0	5.9	1.0	190	II	6.5	13	10.4	3.0	6.2	1.0
159	II	7.4	14	6.0	3.0	5.8	1.0	191	II	4.9	10	5.8	3.0	5.2	0.9

Table S10 Structural features for ECA rules 192–255.

R	Cl	H	E	D	In	A	P	R	Cl	H	E	D	In	A	P
192	I	2.7	6	3.5	2.2	4.9	1.0	224	I	4.9	10	6.6	3.0	5.1	0.9
193	II	4.3	9	4.2	3.0	4.7	1.0	225	III	5.9	12	9.4	3.0	5.2	1.0
194	II	4.1	9	4.7	3.0	4.7	1.0	226	II	6.0	12	8.3	3.0	5.3	1.0
195	III	3.8	8	5.2	2.4	4.7	1.0	227	II	6.5	12	6.9	3.0	5.6	1.0
196	II	5.1	10	6.2	3.0	5.2	1.0	228	II	5.9	12	7.9	3.0	5.5	1.0
197	II	6.6	13	8.2	3.0	5.4	1.0	229	II	6.2	12	6.9	3.0	5.6	1.0
198	II	5.9	12	9.4	3.0	5.6	1.0	230	II	6.2	12	7.8	3.0	5.9	1.0
199	II	5.3	11	7.2	3.0	5.5	1.0	231	II	5.4	12	6.1	3.0	5.5	1.0
200	II	5.5	11	7.0	3.0	5.4	0.9	232	II	6.3	13	10.0	3.0	5.0	0.9
201	II	6.6	13	10.1	3.0	5.5	1.0	233	II	6.7	13	8.3	3.0	5.6	1.0
202	II	6.3	12	9.3	3.0	5.6	1.0	234	I	5.8	12	9.3	3.0	5.0	1.0
203	II	6.0	12	6.5	3.0	5.8	1.0	235	I	6.1	12	8.3	3.0	5.2	1.0
204	II	1.9	5	4.1	2.5	4.5	1.0	236	II	5.5	12	9.5	3.0	5.5	1.0
205	II	5.0	12	9.2	3.0	4.9	1.0	237	II	6.1	12	8.2	3.0	5.7	1.0
206	II	4.6	11	8.2	3.0	5.1	1.0	238	I	3.5	7	6.4	2.5	5.1	1.0
207	II	3.7	8	5.6	2.6	5.0	1.0	239	I	4.4	10	6.7	3.0	5.2	1.0
208	II	4.7	10	6.3	3.0	5.1	1.0	240	II	1.6	4	3.2	2.3	4.5	0.9
209	II	5.9	12	8.6	3.0	5.1	1.0	241	II	5.3	12	9.3	3.0	5.0	1.0
210	II	5.8	12	8.7	3.0	5.3	1.0	242	II	5.4	12	10.1	3.0	5.2	0.9
211	II	5.1	11	6.3	3.0	5.3	1.0	243	II	3.3	7	5.5	2.5	4.5	0.9
212	II	6.1	13	9.0	3.0	5.0	0.9	244	II	5.0	11	8.9	3.0	4.8	1.0
213	II	4.8	10	5.4	3.0	5.4	1.0	245	II	3.5	7	5.1	2.5	4.7	0.9
214	II	6.4	13	8.1	3.0	5.9	0.9	246	II	6.2	13	9.5	3.0	5.6	1.0
215	II	6.1	12	4.7	3.0	5.5	1.0	247	II	5.2	11	4.8	3.0	5.6	1.0
216	II	6.4	13	9.8	3.0	5.6	1.0	248	I	5.0	11	8.2	3.0	5.4	1.0
217	II	6.4	12	6.8	3.0	5.9	1.0	249	I	5.9	12	7.6	3.0	5.4	1.0
218	II	6.4	12	8.6	3.0	5.4	0.9	250	I	3.2	7	6.4	2.4	4.9	0.9
219	II	5.5	12	6.8	3.0	5.7	1.0	251	I	4.7	10	6.7	3.0	5.3	0.9
220	II	4.8	11	8.0	3.0	5.4	1.0	252	I	3.5	8	6.6	2.4	4.9	1.0
221	II	3.2	7	5.1	2.4	4.8	0.9	253	I	4.7	10	6.7	3.0	5.1	1.0
222	II	6.0	13	8.5	3.0	5.7	1.0	254	I	4.4	11	7.7	3.0	4.9	1.0
223	II	4.9	11	5.0	3.0	5.0	1.0	255	I	1.8	5	3.5	2.5	4.3	0.9

Table S11 Feature groups for clustering analysis.

Group	Source	Features
<i>Information-Theoretic</i>		
Structural	FI Analysis	I_S mean and standard deviation
Integration	FI Analysis	Φ mean and standard deviation
Behavioral	FI Analysis	I_B, I_E
Transfer	FI Analysis	T_E mean and standard deviation
Compression	FI Analysis	MDL mean and standard deviation
<i>Graph Structure</i>		
Size	GraphML	Hidden nodes, edges
Depth	GraphML	DAG depth
Inputs	GraphML	Input usage count, uses all inputs
Activations	GraphML	Activation function counts
Aggregation	GraphML	Product aggregation usage
Weights	GraphML	Weight statistics
Topology	GraphML	Centrality, degree statistics

Table S12 Top 10 clustering results (information-theoretic features only) by macro cohesion.

Rank	Features	Macro	C1	C2	C3	C4
1	$I_S + \Phi + I_B + \text{MDL}$	94.8%	91%	99%	89%	100%
2	$\Phi + I_B$	91.9%	91%	99%	78%	100%
3	$I_S + \Phi + \text{MDL}$	90.3%	100%	98%	78%	86%
4	$\Phi + T_E$	89.9%	87%	95%	78%	100%
5	$I_B + T_E$	88.8%	61%	94%	100%	100%
6	Φ only	86.9%	100%	95%	67%	86%
7	$I_S + \Phi + I_B$	86.2%	61%	95%	89%	100%
8	$\Phi + I_B + \text{MDL}$	86.2%	61%	95%	89%	100%
9	$I_S + \Phi$	85.4%	91%	95%	67%	86%
10	$\Phi + \text{MDL}$	84.7%	87%	95%	67%	86%

Table S13 Feature importance: frequency in top 50 clustering results.

Feature Group	FI Only	All Features
Integration (Φ)	66%	–
Behavioral (I_B, I_E)	64%	–
Structural (I_S)	42%	42%
Compression (MDL)	42%	–
Transfer (T_E)	42%	–
Aggregation	–	86%
Weights	–	72%
Size	–	52%
Depth	–	52%
Topology	–	50%
Activations	–	30%
Inputs	–	4%

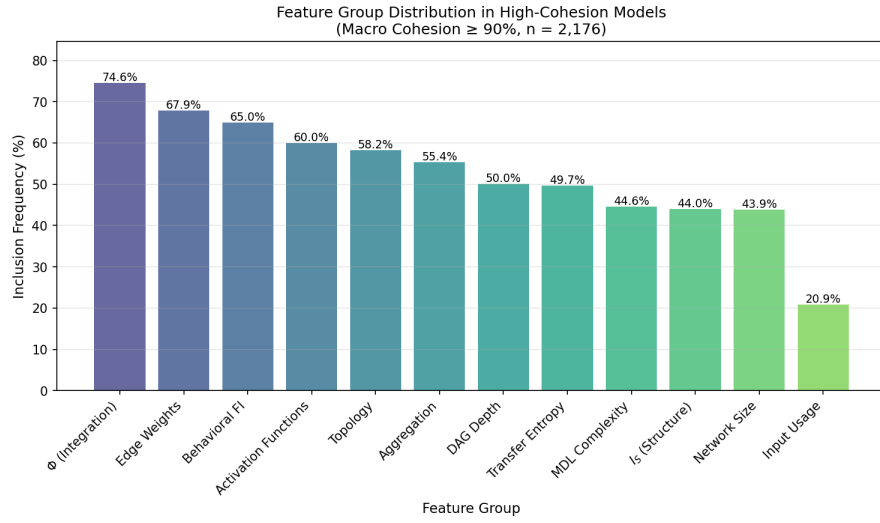


Fig. S1 Feature importance for Wolfram class prediction. Integration (Φ) emerges as the most discriminative feature, appearing in 74.6% of top clustering results. This aligns with the key finding that integration distinguishes algebraic complexity (XOR rules) from computational complexity (Class IV rules).

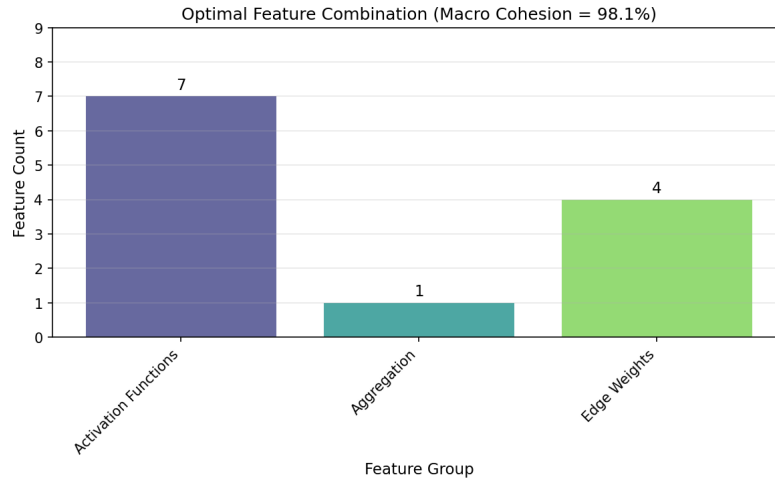


Fig. S2 Optimal feature combination for Wolfram class clustering. The best-performing combination uses Structural Information (I_S), Integration (Φ), Behavioral Information (I_B), and Minimum Description Length (MDL), achieving 94.8% macro cohesion.

Table S14 XOR rules exhibit maximum integration despite minimal structure.

Rule	Function	Class	Hidden	I_S (bits)	Φ
102	$L \oplus C$	III	2.5	81	0.633
90	$L \oplus R$	III	2.6	83	0.467
60	$C \oplus R$	III	2.8	85	0.367
105	$L \oplus C \oplus R$	III	5.3	170	0.200
110	Turing-complete	IV	6.5	222	0.007
41	Complex	IV	7.7	262	0.000
30	Chaotic	III	6.1	214	0.000

Table S15 Correlations between Integration (Φ) and architectural variance across 256 ECA rules. All correlations are significant ($p < 0.001$), supporting the hypothesis that integrated systems have constrained solution spaces.

Variance Measure	Pearson r	p -value
σ_{hidden} (hidden node count)	-0.234	1.6×10^{-4}
σ_{I_S} (Structural Information)	-0.223	3.3×10^{-4}
σ_{conns} (connection count)	-0.250	5.4×10^{-5}

Table S16 Architectural variance by Wolfram class. Class IV rules show highest variance despite lowest mean integration, supporting the modularity-evolvability connection.

Class	n	$\text{mean}(\Phi)$	$\text{mean}(\sigma_{\text{hidden}})$	$\text{mean}(\sigma_{I_S})$
I	24	0.015	1.64	148.1
II	185	0.015	1.87	175.8
III	41	0.065	1.94	188.5
IV	6	0.017	2.32	225.7

Table S17 Individual rule variance. High- Φ rules (90, 102) show low variance; low- Φ Class IV rules (41, 54) show high variance.

Rule	Class	Φ	σ_{hidden}	σ_{I_S}	Interpretation
102	III	0.633	0.85	80.3	Constrained (XOR)
90	III	0.467	0.66	59.0	Constrained (XOR)
60	III	0.367	0.73	68.1	Constrained (XOR)
105	III	0.200	2.14	219.5	Moderate
110	IV	0.007	2.20	221.6	Diverse (modular)
41	IV	0.000	3.79	384.1	Very diverse
54	IV	0.000	3.13	300.4	Very diverse
124	IV	0.000	1.69	161.0	Diverse
30	III	0.000	2.32	214.7	Diverse (chaotic)