

Complexity: Several Questions Wearing One Word

IDE-016 · Fundamental Ideas

Concept essays for The Universe and Its Observers

Author: Juan Segurola

Version: v0.5 EN · 19 September 2026

Status: Publication version

Article form: Concept essay

Scale / context: Strings, algorithms, networks and ecosystems

Boundary: Complexity as a family of measures and what each one measures. Novel collective organization belongs to IDE-007; limits on knowledge and prediction belong to IDE-017

Related works: IDE-017, IDE-007, IDE-006

Epistemic profile: Established knowledge with interpretive boundaries

Licence at publication: All Rights Reserved

Article Abstract

There is no single quantity called complexity. Dozens of measures exist and they answer different questions: how long a description must be, how much computation an answer costs, how much history a structure encodes, how strongly parts depend on one another.[1] These come apart in ways that matter. A random string is maximally hard to compress and has no organization at all, so description length measures randomness rather than structure, and any measure that peaks on noise is measuring the wrong thing for most purposes.[2,4] This article separates the main families, shows why the measure has to be chosen with the question, and closes on a result that reversed an intuition: adding connections to a large system can make it less stable, not more.[5]

In 60 Seconds

Ask what makes something complex and you get several incompatible answers, all defensible. Algorithmic incompressibility characterizes randomness; logical depth separately distinguishes shallow random or simple objects from objects with deep computational histories.[2] Computational complexity asks not how long the description is but how much work the answer costs as the problem grows, and the classification of problems by that cost is where the hardest open question in the subject lives.[3] Logical depth asks a third thing: how much computation would be needed to generate the object from its shortest description, which is high for structures that took a long process to produce and low for both crystals and noise.[4] Effective complexity separates the regular part of a description from its random part and measures only the former.[6] Each is a real quantity; none is complexity itself, and using one where another is meant is the usual source of confusion.

Why This Question Matters

Complexity is invoked to explain why something is hard to predict, hard to build, or resistant to simple description, and those are different properties that can occur separately. A protein and a shuffled deck

are not complex in the same sense, and a claim that a system is too complex to model is empty until it says which measure is large and why that blocks the inference being attempted. The word also carries a false implication that has done real damage in ecology and in systems design, namely that more parts and more connections buy robustness.

1. No Single Scalar

Lloyd's list of measures is deliberately titled nonexhaustive, and its point is structural rather than encyclopaedic. The measures cluster around three questions: how hard is it to describe, how hard is it to create, and to what degree is it organized.[1] Those are different questions, and no argument shows they must have a common answer.

This matters because complexity claims usually smuggle in a comparison. Saying that a cell is more complex than a crystal is defensible under most measures; saying that one ecosystem is more complex than another usually needs the measure named before it means anything. Inference: a complexity claim is incomplete unless it identifies the measure being used.

The practical demand is therefore modest and rarely met. State which quantity is being claimed to be large, over what description of the system, and why that quantity is the one relevant to the conclusion being drawn. Most uses of the word fail the first clause.

Where That Leaves Us

Complexity measures form distinct families organized around different questions, including difficulty of description, difficulty of creation and degree of organization.[1] The first family, description length, has a formal version whose behaviour is instructive and initially surprising.

2. Description Length, and Its Trap

Kolmogorov defined the complexity of a string as the length of the shortest program that produces it on a universal machine, which makes precise the idea that a regular object has a short specification.[2] A string of a million zeros is compressible to a short instruction; a string with no exploitable pattern is not.

The definition has two well-known properties that shape everything built on it. It depends on the choice of universal machine, though only up to an additive constant, so comparisons of long strings are meaningful and comparisons of short ones are not.[2] And it is not computable: no general procedure finds the shortest program, so the quantity is a theoretical reference rather than something measured.[7]

Algorithmic incompressibility characterizes randomness; logical depth separately distinguishes shallow random or simple objects from objects with deep computational histories.[2,4,6]

Where That Leaves Us

Algorithmic incompressibility characterizes randomness; logical depth separately distinguishes shallow random or simple objects from objects with deep computational histories. It is also uncomputable and machine-dependent up to a constant. Measures that avoid peaking on noise have to ask about something other than the length of a description.

3. Effort and History, Not Just Length

Computational complexity asks how the resources needed to solve a problem grow with its size. Cook's 1971 result identified a class of problems to which every problem in a broad family can be reduced, which made it possible to speak of the hardest problems in that family rather than of individual difficult cases.[3] Whether that family can be solved efficiently is the central open question of the field.

Bennett's logical depth asks a different question again: how much computation is required to produce the object from its shortest description.[4] A crystal is shallow because a short program produces it quickly; random noise is shallow because the shortest program is the string itself, printed immediately. What is deep is an object whose short description exists but takes a long computation to unfold, which is a reasonable formal image of something that took a long history to produce.

Gell-Mann and Lloyd's effective complexity takes the complementary route. It separates a description into the part that captures regularities and the part that is incompressible detail, and measures only the regular part.[6] Noise then scores low because almost all of its description is the random component, which is the behaviour the intuitive notion demands.

Where That Leaves Us

Computational complexity measures the cost of an answer, logical depth measures the computation needed to unfold a short description, and effective complexity measures only the regular part of a description. Each avoids the noise trap differently. All three, though, are relative to how the object was described in the first place.

4. Complexity of What, Exactly

Complexity measures include description-based and computational or generative types; effective complexity specifically depends on which regularities are identified.[1,6] A gas is trivially describable by two thermodynamic variables and impossibly complicated as a list of molecular positions, and both descriptions are of the same gas.

The discipline is the same as elsewhere in this series. Name the description, name the measure, and check that the conclusion depends on the quantity actually computed.

Where That Leaves Us

Complexity measures include description-based and computational or generative types; effective complexity specifically depends on which regularities are identified.[1,6] That bears on a widely held belief about what complexity buys.

5. More Parts Is Not More Robust

It was once a working assumption in ecology that more diverse and more connected communities are more stable. May analysed randomly assembled model communities and found the opposite tendency: as the number of species, the connectance and the interaction strength increase, the probability that the equilibrium is locally stable falls sharply.[5]

The result is about a specific model class, randomly constructed interaction matrices, and about local stability at an equilibrium, and May was explicit about that. Its force is not that real

ecosystems are unstable but that stability cannot be inferred from complexity alone: whatever stabilizes real communities has to be a feature of how they are structured, not of how many parts they have.

The general lesson transfers. Adding components and couplings to any system enlarges the space of ways it can fail, and robustness comes from architecture rather than from count. This is the same conclusion that arrives from the feedback side, where stability depends on the arrangement of gains and delays rather than on the number of loops.

Where That Leaves Us

May showed that in randomly assembled model communities greater size, connectance and interaction strength reduce the chance of local stability, so complexity does not buy robustness and whatever stabilizes real systems is structural. The result is bounded to its model class, which is part of reading it correctly. IDE-017 takes up the family of limits that complexity is often blamed for: uncertainty and its sources.

What We Know, and How

- Complexity measures form distinct families organized around different questions, including difficulty of description, difficulty of creation and degree of organization.[1]
- Kolmogorov complexity is the length of the shortest program producing a string, is machine-dependent only up to an additive constant and is uncomputable; algorithmic incompressibility characterizes randomness.[2,7]
- Logical depth measures the computation needed to generate an object from its shortest description and is low for both crystals and noise; effective complexity measures only the regular part of a description.[4,6]
- In randomly assembled model communities, increasing species number, connectance and interaction strength reduces the probability of local stability, so complexity does not by itself confer robustness.[5]

What We Still Don't Know

- Whether the broad family of problems identified by Cook's reduction can be solved efficiently, which remains the central open question of computational complexity.[3]
- Which complexity measure, if any, tracks the intuitive notion of organized structure across physical, biological and social systems rather than within a single domain.
- How far May's stability result extends to interaction structures that are not randomly assembled, which is where real ecological and engineered networks live.

What Could Change Our Minds

- A proof settling whether the family of problems Cook identified admits efficient solution would resolve the central question of the field and reorganize the classification of computational difficulty.[3]

- A computable measure that assigned low values to both random and perfectly regular strings, high values to organized structure, and agreed with logical depth and effective complexity where those are calculable, would replace the current plurality of measures.[1,4,6]
- Empirical evidence that ecological communities with higher connectance are systematically more stable under comparable perturbation, with interaction structure controlled, would require revision of how May's result is generalized.[5]

If You Remember Only One Thing

Complexity comprises distinct families of measures addressing description, creation and organization.
[1,6]

Related Articles

- IDE-017 — Uncertainty
- IDE-007 — Emergence
- IDE-006 — Scale

References

- [1] Seth Lloyd. Measures of Complexity: A Nonexhaustive List. IEEE Control Systems Magazine 21(4), 7-8 (2001). DOI 10.1109/37.969131.
- [2] A. N. Kolmogorov. Three Approaches to the Quantitative Definition of Information. Problems of Information Transmission 1(1), 1-7 (1965).
- [3] Stephen A. Cook. The Complexity of Theorem-Proving Procedures. Proceedings of the Third Annual ACM Symposium on Theory of Computing, 151-158 (1971). DOI 10.1145/800157.805047.
- [4] Charles H. Bennett. Logical Depth and Physical Complexity. In: Rolf Herken (ed.), The Universal Turing Machine: A Half-Century Survey, Oxford University Press, 227-257 (1988).
- [5] Robert M. May. Will a Large Complex System be Stable? Nature 238, 413-414 (1972). DOI 10.1038/238413a0.
- [6] Murray Gell-Mann; Seth Lloyd. Information Measures, Effective Complexity, and Total Information. Complexity 2(1), 44-52 (1996).
- [7] Ming Li; Paul Vitányi. An Introduction to Kolmogorov Complexity and Its Applications, 4th edition. Springer (2019), chapter 2.