Algorithmic Information Dynamics

Hector Zenil

Algorithmic Dynamics Lab / Karolinska Institute
Stockholm, Sweden &
Oxford Immune Algorithmics, Oxford, U.K.

Algorithmic Dynamics Lab

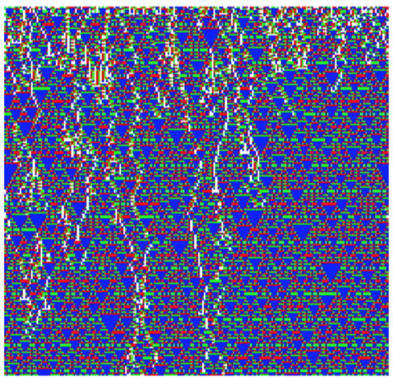
Algorithmic Nature Group

Oxford Immune Algorithmics

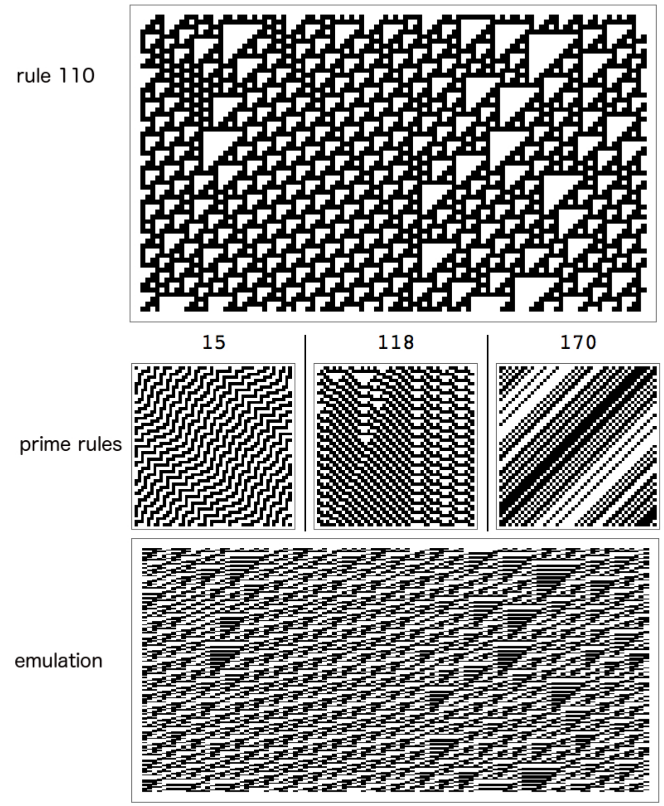
Oxford University Innovation

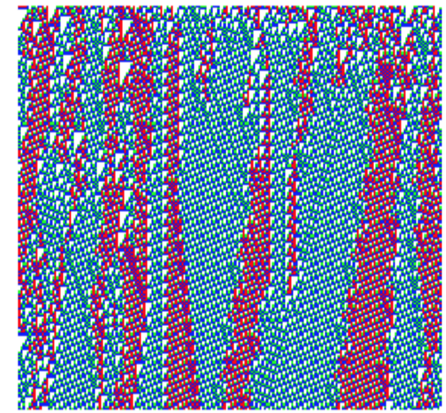
University of Oxford

Composition as a source of unbounded complexity from computation



Composition of ECA rules 50 ∘ 37 with colour remapping leading to a 4-colour Turing universal CA emulating rule 110.





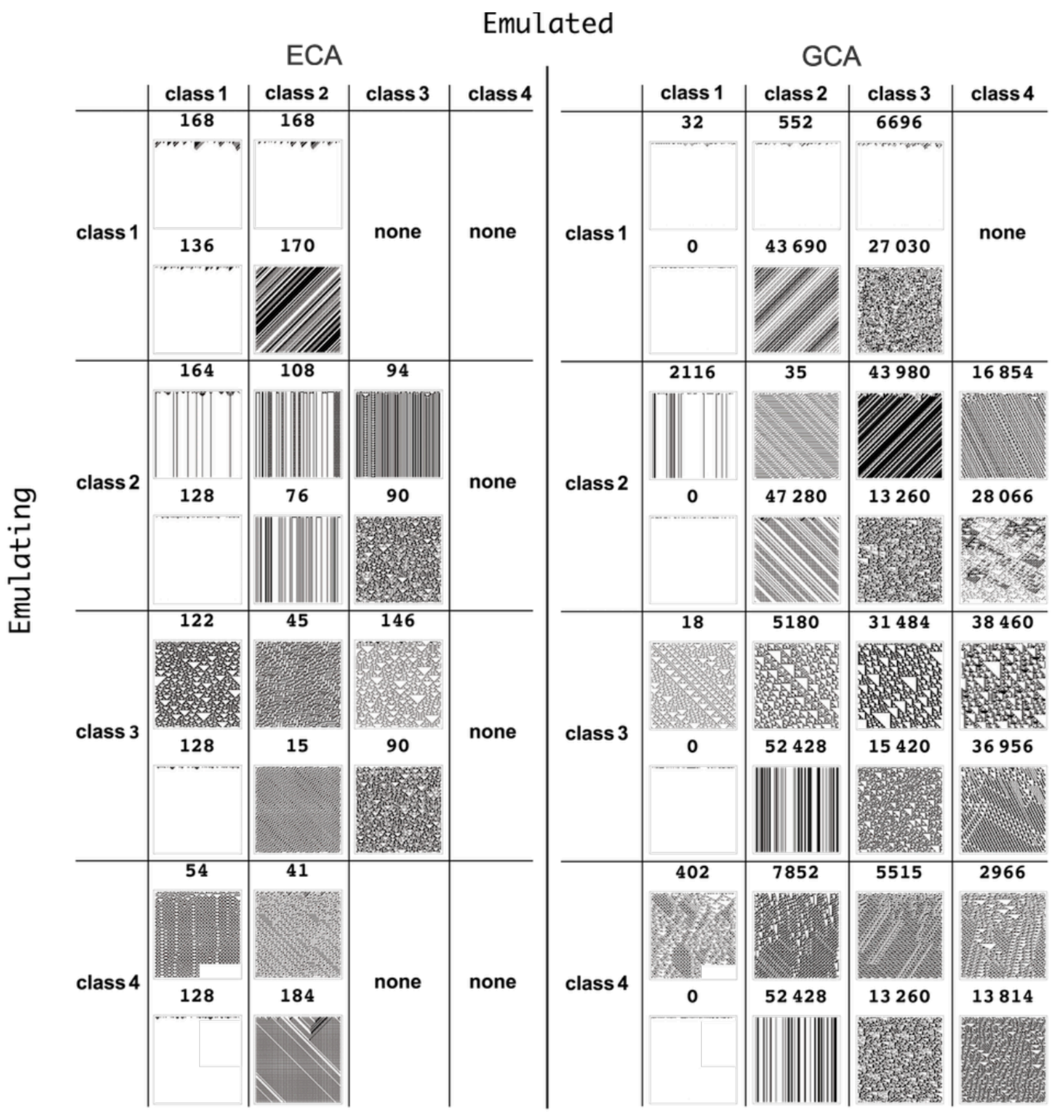
Composition of ECA rules 170 ∘ 15 ∘ 118 with colour re-mapping mapping leading to a 4-colour Turing universal CA emulating rule 110.

Boolean composition of prime ECA rules 15, 118 and 170 simulates ECA rule 110

Proving the power of interactions to produce unbounded complexity (and random-looking behaviour)

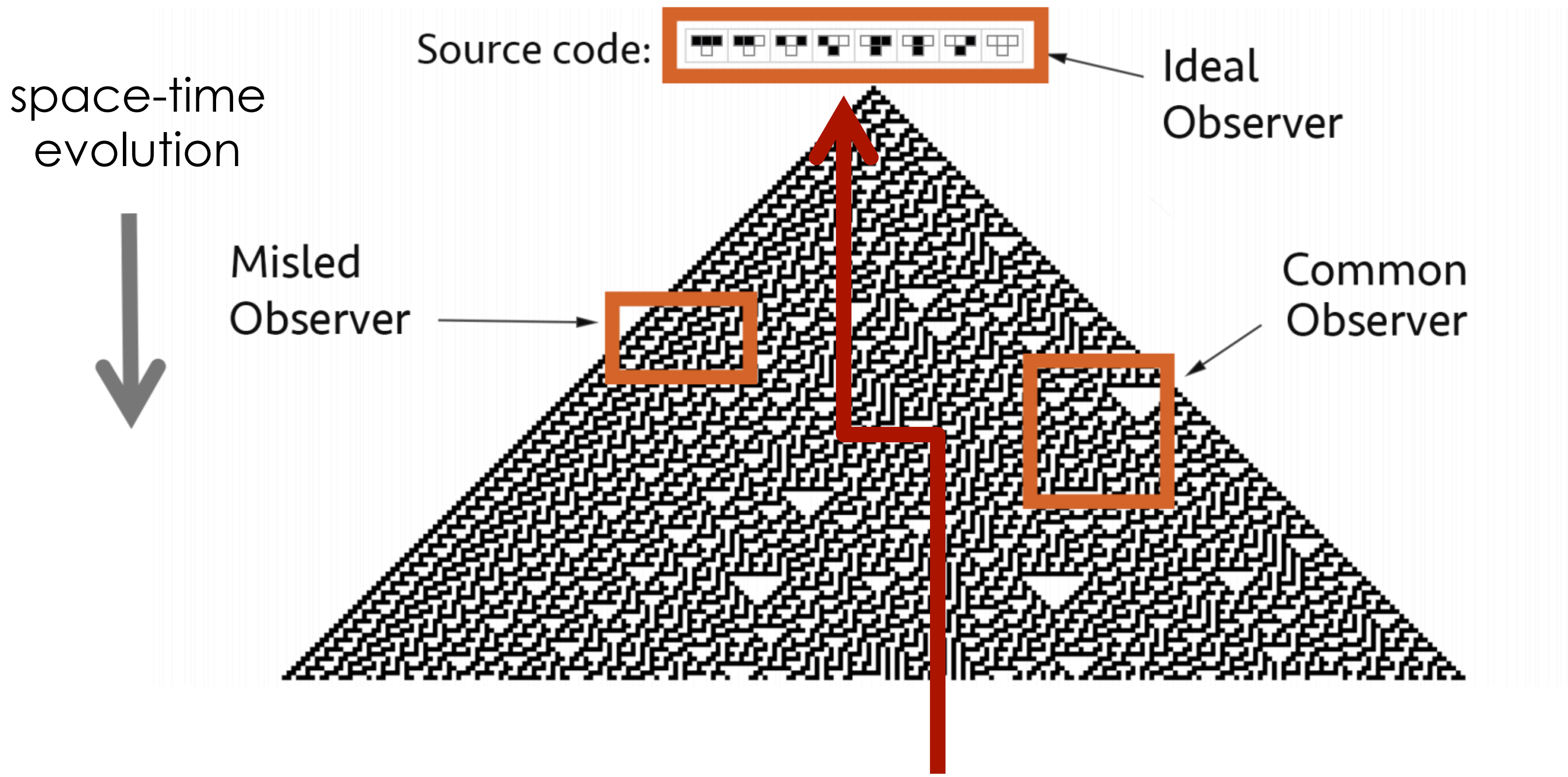
J. Riedel and H. Zenil, Rule Primality, Minimal Generating Sets and Turing-Universality in the Causal Decomposition of Elementary Cellular Automata, Journal of Cellular Automata, vol. 13, pp. 479–497, 2018

Pervasive Turing universality?



J Riedel, H. Zenil, Cross-boundary Behavioural Reprogrammability Reveals Evidence of Pervasive Turing Universality, International Journal of Unconventional Computing, vol 13:14-15 pp. 309-357, 2018.

The hacker view of causality



Can we infer these rule using classical information theory?
(computational mechanics without the stochastic part)

Correlation v Causation

Entropy can only "see" statistical regularities

Thue-Morse sequence: 01101001100101101001011001101001
Segment of π in binary: 00100100001111110110101010001000

Definition

Kolmogorov(-Chaitin) complexity (1965,1966):

$$K_U(s) = min\{|p|, U(p) = s\}$$

The power of K (algorithmic randomness)

Martin Löf proves (1966) that K captures all possible computable properties, and so a random string s is random if it is typical in the sense that there is no property that shortens any description of s. A string s is random if $K(s)$ (in bits) $\sim |s|$.

Correlation v Causation

Entropy can only "see" statistical regularities

Thue-Morse sequence: 01101001100101101001011001101001

Segment of π in binary: 00100100001111110110101010001000

Definition

Kolmogorov(-Chaitin) complexity (1965,1966):

$$K_U(s) = min\{|p|, U(p) = s\}$$

The power of K (algorithmic randomness)

Convergence of definitions!

Martin Löf proves (1966) that K captures all possible computable properties, and so a random string s is random if it is typical in the sense that there is no property that shortens any description of s. A string s is random if $K(s)$ (in bits) $\sim |s|$.

Correlation v Causation

Entropy can only "see" statistical regularities

Thue-Morse sequence: 01101001100101101001011001101001

Segment of π in binary: 00100100001111110110101010001000

Definition

Semi-computable!!!

Kolmogorov(-Chaitin) complexity (1965,1966):

$$K_U(s) = min\{|p|, U(p) = s\}$$

The power of K (algorithmic randomness)

Convergence of definitions!

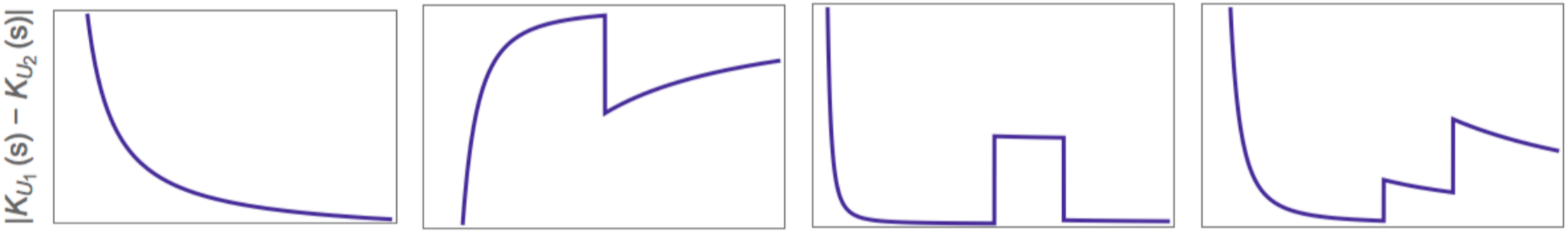
Martin Löf proves (1966) that K captures all possible computable properties, and so a random string s is random if it is typical in the sense that there is no property that shortens any description of s. A string s is random if $K(s)$ (in bits) $\sim |s|$.

Estimating K in practice

Do we measure K with programming language or universal TM U_1 or U_2?
The *Invariance Theorem*:

$$|K_{U_1}(s) - K_{U_2}(s)| < c_{U_1,U_2}$$

It is not relevant in the limit, the difference is a constant that vanishes the longer the strings.



Rate of convergence of K and the behaviour of c with respect to $|s|$

The Invariance theorem in practice is a negative result

The constant involved can be arbitrarily large, the theorem tells nothing about the convergence. Any estimating method of K is subject to it.

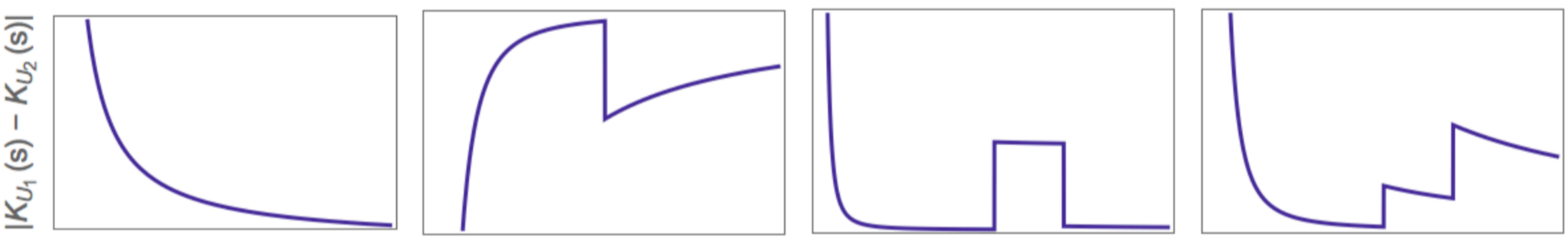
Estimating K in practice

In principle: more serious that uncomputability!

Do we measure K with programming language or universal TM U_1 or U_2?
The *Invariance Theorem*:

$$|K_{U_1}(s) - K_{U_2}(s)| < c_{U_1,U_2}$$

It is not relevant in the limit, the difference is a constant that vanishes the longer the strings.



Rate of convergence of K and the behaviour of c with respect to $|s|$

The Invariance theorem in practice is a negative result

The constant involved can be arbitrarily large, the theorem tells nothing about the convergence. Any estimating method of K is subject to it.

Algorithmic Information Theory in Molecular Biology

Superficially landmark results turn out to reproduce trivial results in structural biology:

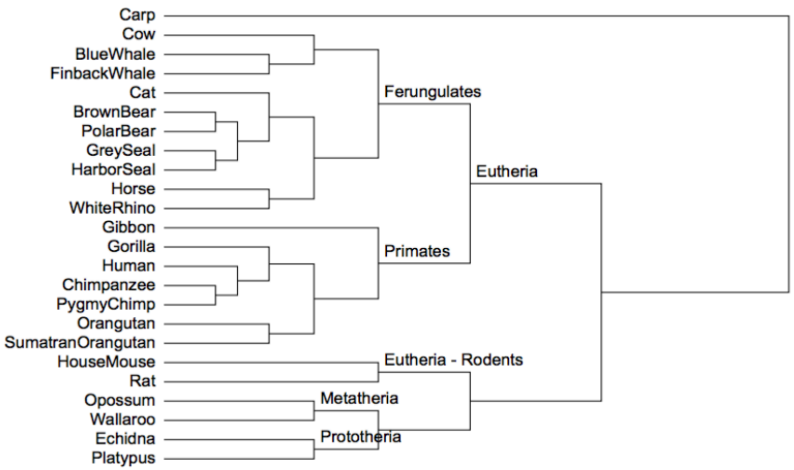


Figure: Each species' average GC-content lies on the curve determining its place in the phylogenetic space.

[R. Cilibrasi and P.M.B. Vitányi, Clustering by Compression (2005)]

Can we do better than practical lossless compression?

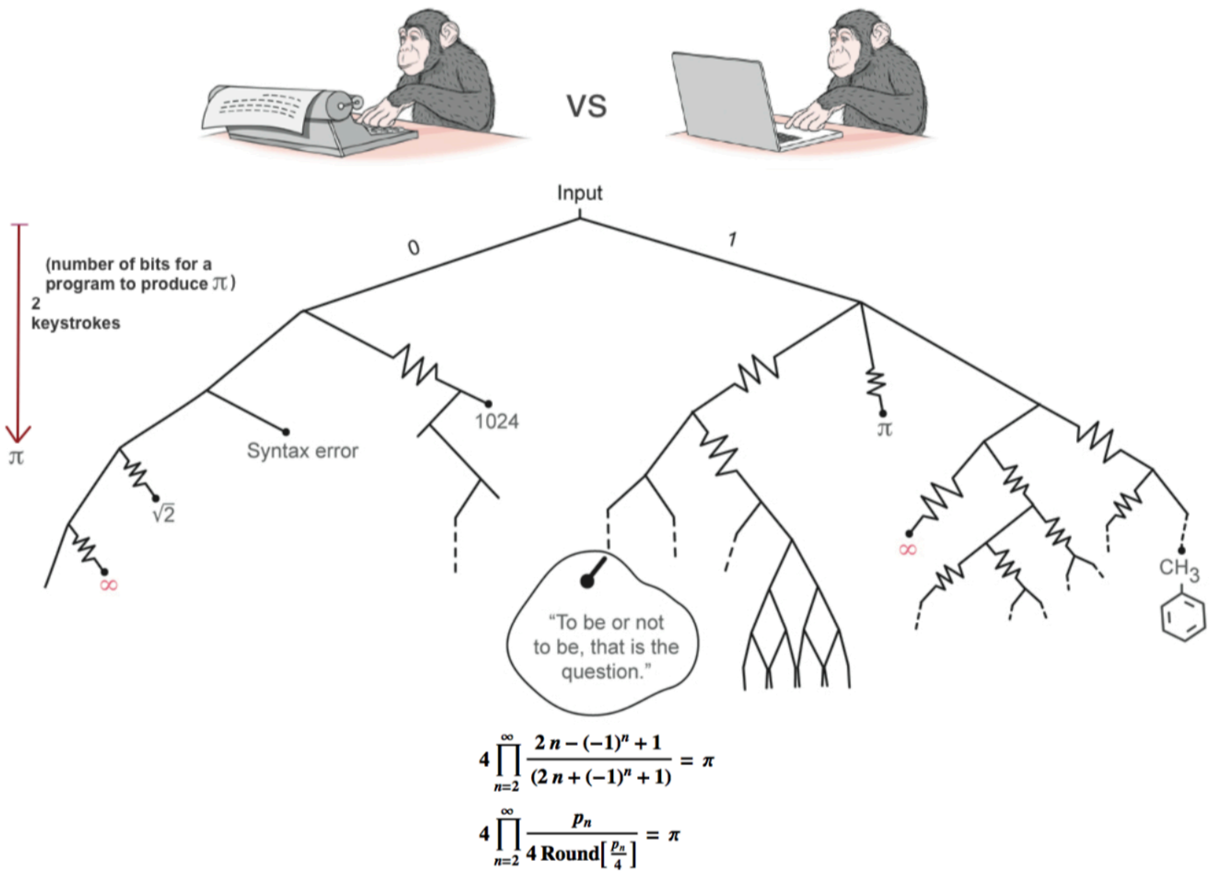


Figure: (originally Emile Borel's *infinite monkey theorem*): A monkey on a computer has greater chances to produce structure than a monkey on a typewriter.

[Inspired by a sketch from C. Bennett]

Algorithmic Probability and Coding theorem

$$m(x) = \sum_{p:U(p)=x} 1/2^{|p|}$$

$$K(s) = -\log_2 m(s) + O(1)$$

J.-P. Delahaye and H. Zenil, Numerical Evaluation of the Complexity of Short Strings: A Glance Into the Innermost Structure of Algorithmic Randomness
Applied Mathematics and Computation 219, pp. 63-77, 2012.

Computability, Algorithmic Complexity & Causality?

(Un)Computability mediates in the challenge of algorithmic causality by way of Algorithmic Information Theory:

$$K(s) = \min\{ p \mid U(p) = s\}$$

Observation

Method

Most likely mechanism according to Occam's

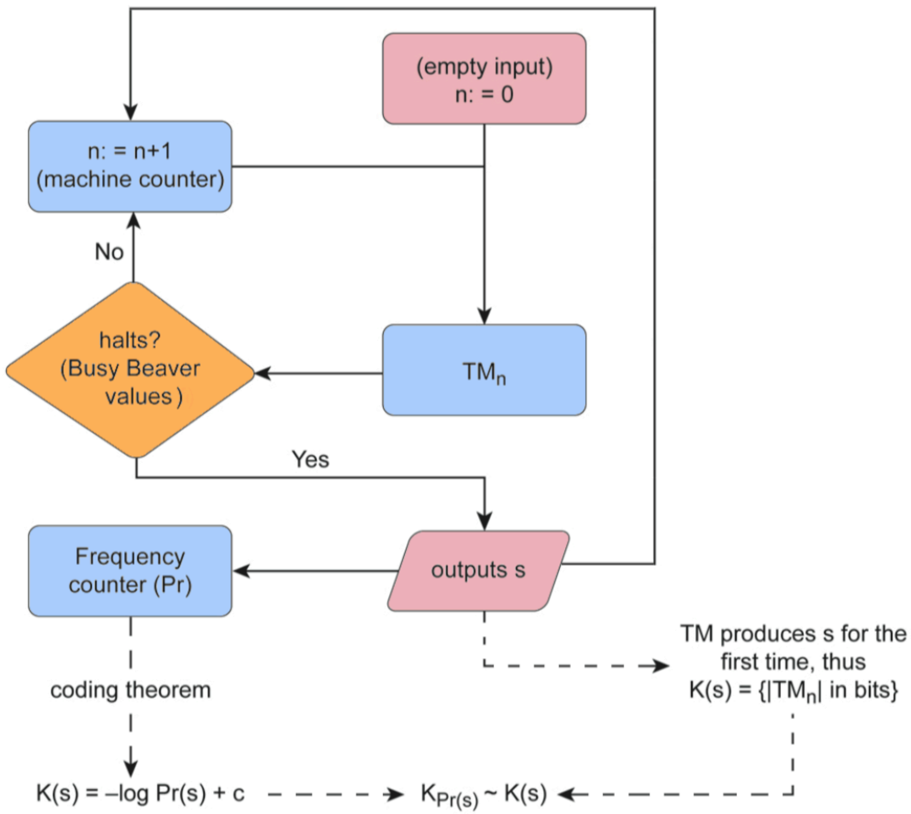
Generating mechanism

Constructor

Compatible with the observation

Can be approximated from above (lower semi-computable)

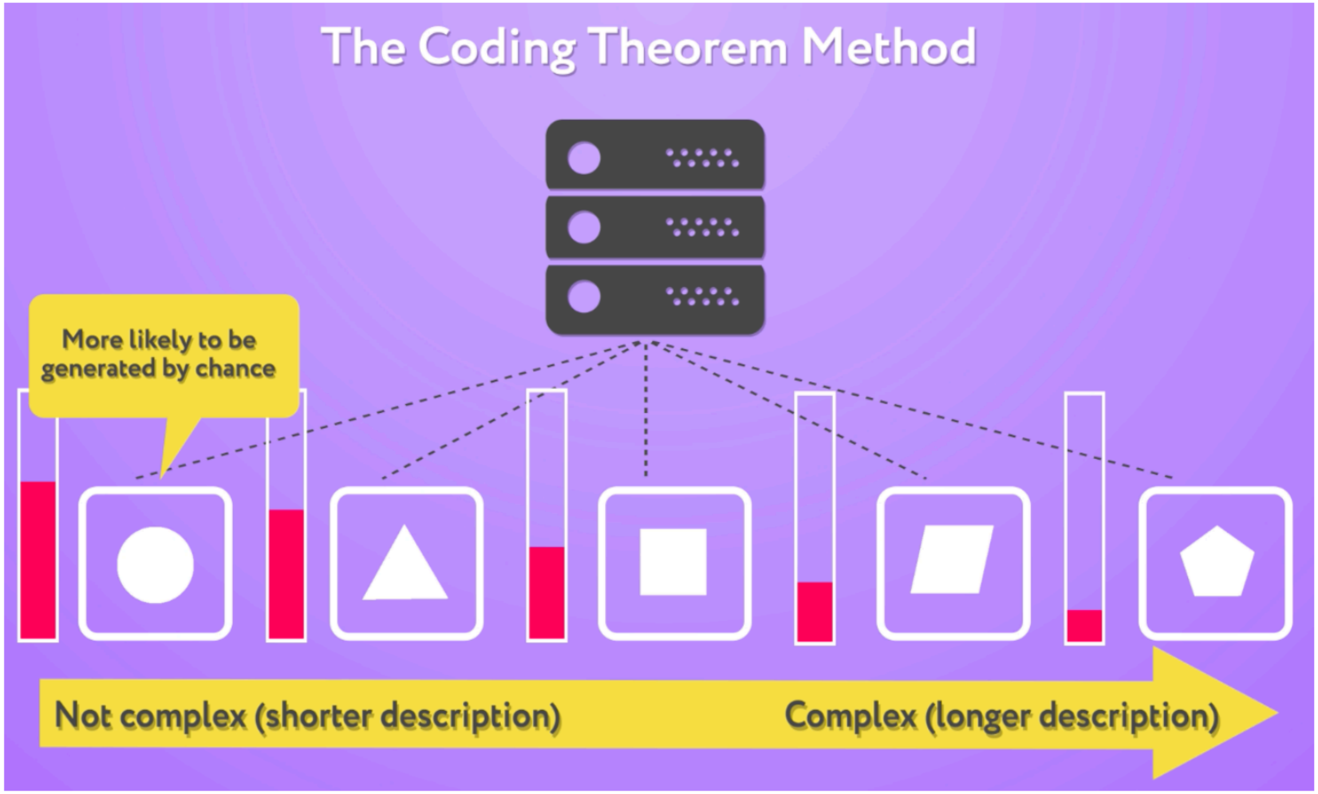
The AP approach to K: The Coding Theorem Method (CTM)



[Soler, Zenil et al, PLoS ONE (2014)]

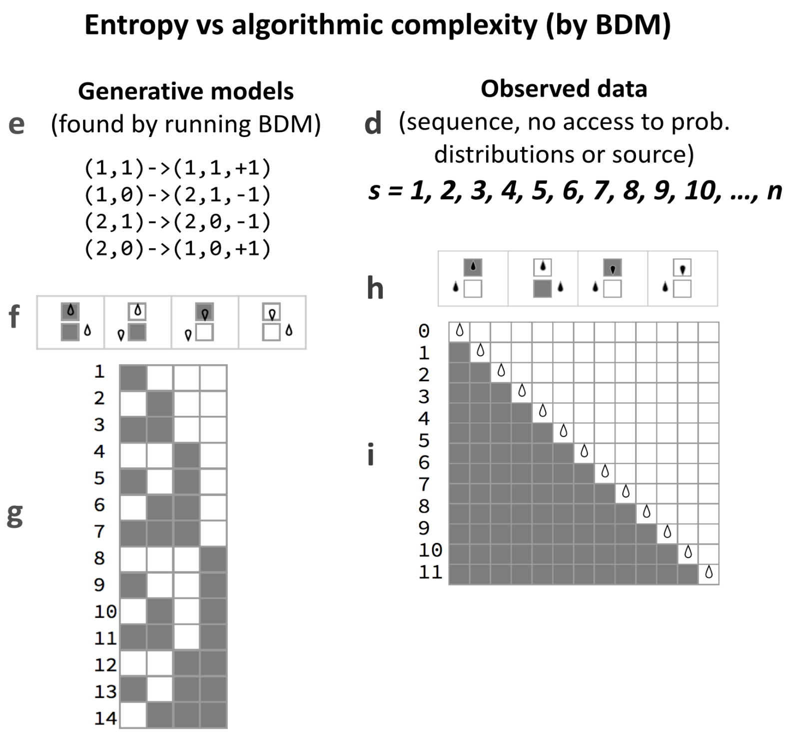
Changing the underlying computational model the distribution remains stable

The Coding Theorem Method (CTM)



J.-P. Delahaye and H. Zenil, Numerical Evaluation of the Complexity of Short Strings: A Glance Into the Innermost Structure of Algorithmic Randomness
Applied Mathematics and Computation 219, pp. 63-77, 2012.

Finding Generative Models



H. Zenil, N.A. Kiani, A. Zea, J. Tegnér, Causal Deconvolution by Algorithmic Generative Models
Nature Machine Intelligence, vol 1(1), pp 58-66, 2019.

The Coding Theorem Method (CTM): A model generator

$$CTM(s) = \frac{\#\text{ of times that a machine } (n, 2) \text{ produces } s}{\#\text{ of machines in } (n, 2)}$$

$$CTM(s) \sim -\log_2 D(n,\ 2)(s)$$

We are less interested by the output real numbers than by the set of (non-necessarily) minimal length programs (candidate models explaining s)

J.-P. Delahaye and H. Zenil, Numerical Evaluation of the Complexity of Short Strings: A Glance Into the Innermost Structure of Algorithmic Randomness
Applied Mathematics and Computation 219, pp. 63-77, 2012.

Significance of Algorithmic Probability

- R. Solomonoff demonstrates that AP is an optimal universal inference method (presented at the Dartmouth Conference 1956 considered the starting point of AI).
- Walter Kirchherr. The Miraculous Universal Distribution. The Mathematical Intelligencer, 1997.

More recently. Marvin Minsky:

> *It seems to me that the most important discovery since Gödel was the discovery by Chaitin, Solomonoff and Kolmogorov of the concept called Algorithmic Probability*
>
> ...
>
> *it should be possible to make practical approximations to the Chaitin, Kolmogorov, Solomonoff theory that would make better predictions than anything we have today. Everybody should learn all about that and spend the rest of their lives working on it.*
>
> *Marvin Minsky Panel on The Limits of Understanding World Science Festival NYC, Dec 14, 2014*

Emergence of the Universal Distribution

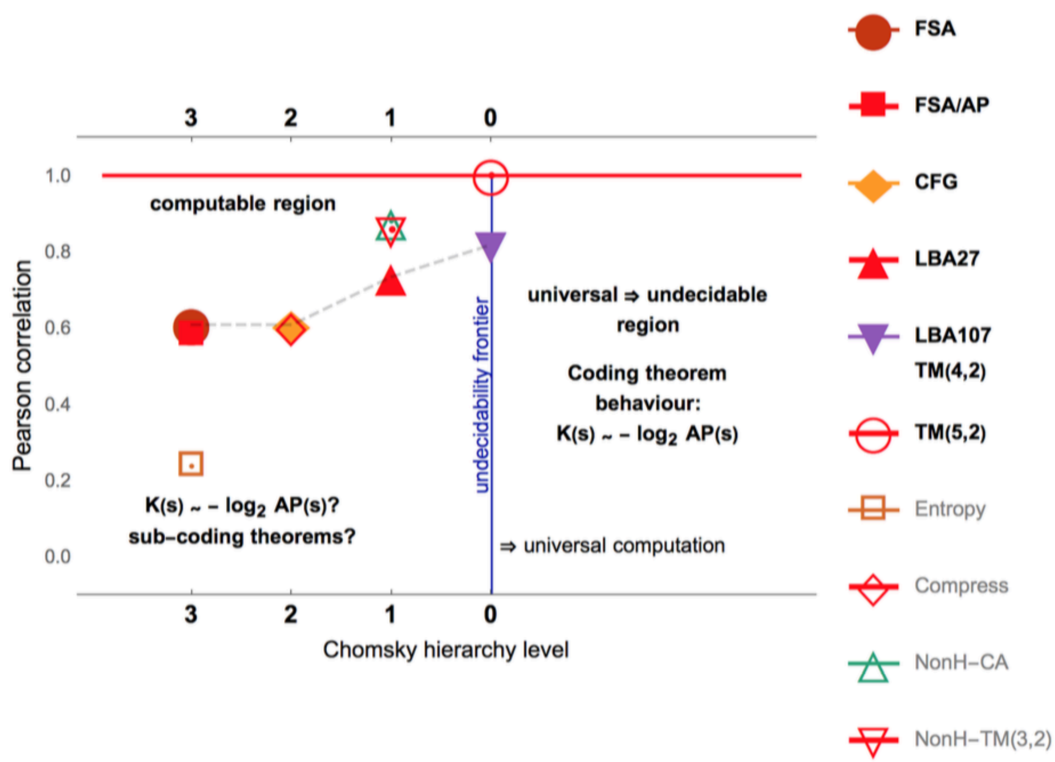


Figure: Emergence of the Universal Distribution. Algorithmic Probability as a function of computational power: Increasing monotonic.

H. Zenil et al., Coding-theorem Like Behaviour and Emergence of the Universal Distribution from Resource-bounded Algorithmic Probability, International Journal of Parallel Emergent and Distributed Systems (2018)

Resources needed for calculation of D

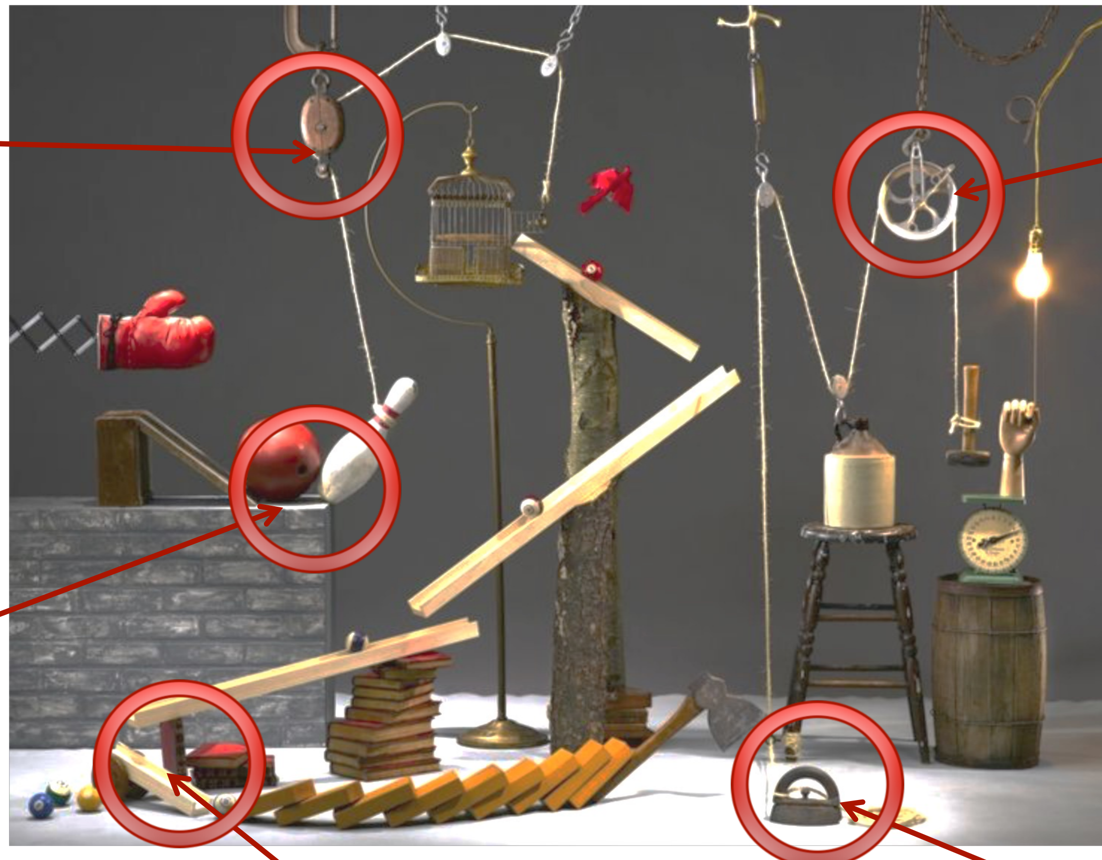
Table: Letter code: F full space, S sample, $R(n,m)$ reduced enumeration. Time is given in seconds (s), hours (h) and days (d).

(n,m)	**Calculation**	**Number of Machines**	**Time**
(2,2)	$F-$ (6 steps)	$\lvert R(2,2)\rvert = 2000$	0.01 s
(3,2)	$F-$ (21)	$\lvert R(3,2)\rvert = 2\,151\,296$	8 s
(4,2)	$F-$ (107)	$\lvert R(4,2)\rvert = 3\,673\,320\,192$	4 h
$(4,2)_{2D}$	$F_{2D}-$ (1500)	$\lvert R(4,2)_{2D}\rvert = 315\,140\,100\,864$	252 d
(4,4)	S (2000)	334×10^9	62 d
(4,5)	S (2000)	214×10^9	44 d
(4,6)	S (2000)	180×10^9	41 d
(4,9)	S (4000)	200×10^9	75 d
(4,10)	S (4000)	201×10^9	87 d
(5,2)	$F-$ (500)	$\lvert R(5,2)\rvert = 9\,658\,153\,742\,336$	450 d
$(5,2)_{2D}$	S_{2D} (2000)	1291×10^9	1970 d

[Soler-Toscano and Zenil et al. PLoS ONE (2014)]

Block Decomposition Method

0 : 0.250	11110 : 0.0000470	100101 : 1.43×10^{-6}
1 : 0.250	00100 : 0.0000456	101001 : 1.43×10^{-6}
00 : 0.101	11011 : 0.0000456	000011 : 9.313×10^{-7}
01 : 0.101	01010 : 0.0000419	000110 : 9.313×10^{-7}
10 : 0.101	10101 : 0.0000419	001100 : 9.313×10^{-7}
11 : 0.101	01001 : 0.0000391	001101 : 9.313×10^{-7}
000 : 0.0112	01101 : 0.0000391	001111 : 9.313×10^{-7}
111 : 0.0112	10010 : 0.0000391	010001 : 9.313×10^{-7}
001 : 0.0108	10110 : 0.0000391	010010 : 9.313×10^{-7}
011 : 0.0108	01110 : 0.0000289	010011 : 9.313×10^{-7}
100 : 0.0108	10001 : 0.0000289	011000 : 9.313×10^{-7}
110 : 0.0108	00101 : 0.0000233	011101 : 9.313×10^{-7}
010 : 0.00997	01011 : 0.0000233	011110 : 9.313×10^{-7}
101 : 0.00997	10100 : 0.0000233	100001 : 9.313×10^{-7}
0000 : 0.000968	11010 : 0.0000233	100010 : 9.313×10^{-7}
1111 : 0.000968	00011 : 0.0000219	100111 : 9.313×10^{-7}
0010 : 0.000699	00111 : 0.0000219	101100 : 9.313×10^{-7}
0100 : 0.000699	11000 : 0.0000219	101101 : 9.313×10^{-7}
1011 : 0.000699	11100 : 0.0000219	101110 : 9.313×10^{-7}
1101 : 0.000699	000000 : 3.733×10^{-6}	110000 : 9.313×10^{-7}
0101 : 0.000651	111111 : 3.733×10^{-6}	110010 : 9.313×10^{-7}
1010 : 0.000651	000001 : 2.793×10^{-6}	110011 : 9.313×10^{-7}
0001 : 0.000527	011111 : 2.793×10^{-6}	111001 : 9.313×10^{-7}
0111 : 0.000527	100000 : 2.793×10^{-6}	111100 : 9.313×10^{-7}
1000 : 0.000527	111110 : 2.793×10^{-6}	0101010 : 9.313×10^{-7}
1110 : 0.000527	000100 : 2.333×10^{-6}	1010101 : 9.313×10^{-7}
0110 : 0.000510	001000 : 2.333×10^{-6}	001110 : 4.663×10^{-7}
1001 : 0.000510	110111 : 2.333×10^{-6}	011100 : 4.663×10^{-7}
0011 : 0.000321	111011 : 2.333×10^{-6}	100011 : 4.663×10^{-7}
1100 : 0.000321	000010 : 1.863×10^{-6}	110001 : 4.663×10^{-7}
00000 : 0.0000969	001001 : 1.863×10^{-6}	0000010 : 4.663×10^{-7}
11111 : 0.0000969	001010 : 1.863×10^{-6}	0000110 : 4.663×10^{-7}
00110 : 0.0000512	010000 : 1.863×10^{-6}	0100000 : 4.663×10^{-7}
01100 : 0.0000512	010100 : 1.863×10^{-6}	0101110 : 4.663×10^{-7}
10011 : 0.0000512	011011 : 1.863×10^{-6}	0110000 : 4.663×10^{-7}
11001 : 0.0000512	100100 : 1.863×10^{-6}	0111010 : 4.663×10^{-7}
00010 : 0.0000489	101011 : 1.863×10^{-6}	1000101 : 4.663×10^{-7}
01000 : 0.0000489	101111 : 1.863×10^{-6}	1001111 : 4.663×10^{-7}
10111 : 0.0000489	110101 : 1.863×10^{-6}	1010001 : 4.663×10^{-7}
11101 : 0.0000489	110110 : 1.863×10^{-6}	1011111 : 4.663×10^{-7}
00001 : 0.0000470	111101 : 1.863×10^{-6}	1111001 : 4.663×10^{-7}
01111 : 0.0000470	010110 : 1.43×10^{-6}	1111101 : 4.663×10^{-7}
10000 : 0.0000470	011010 : 1.43×10^{-6}	



Rube Goldberg machine

If any part of the whole system (samples) is of high m(x) and low K(x), then that part can be generated by mechanistic/algorithmic means and thus is causal. The lower BDM the more causal.

Block Decomposition Method (BDM)

The following is a *BDM* partitioning example for block size $= 6$ and block overlap $= 1$ as an illustration of the meaning of block size and block overlap in the estimation of a complexity of a long string:

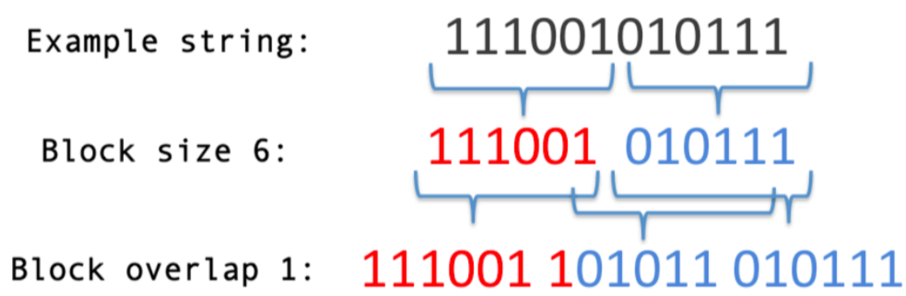
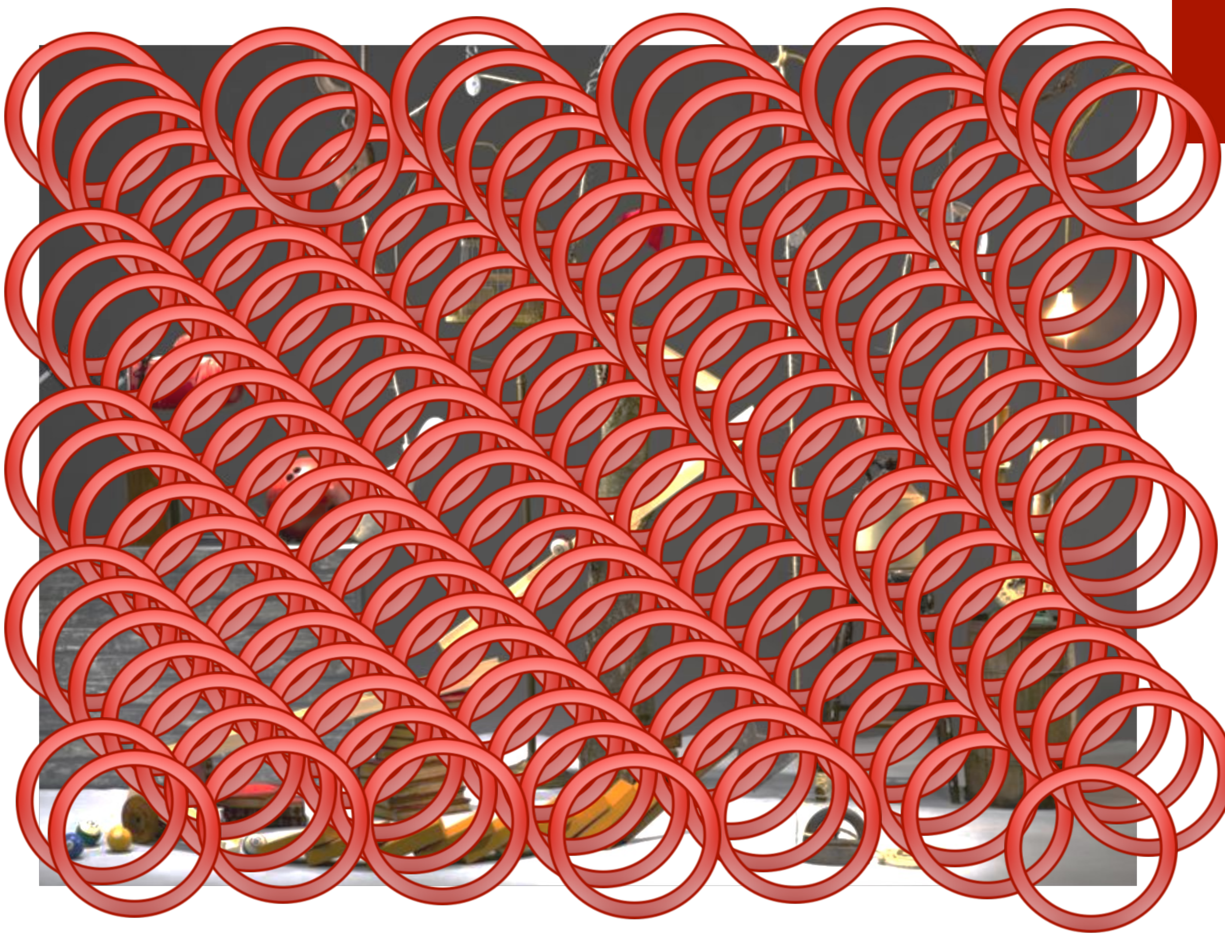


Figure: Block decomposition method (BDM).

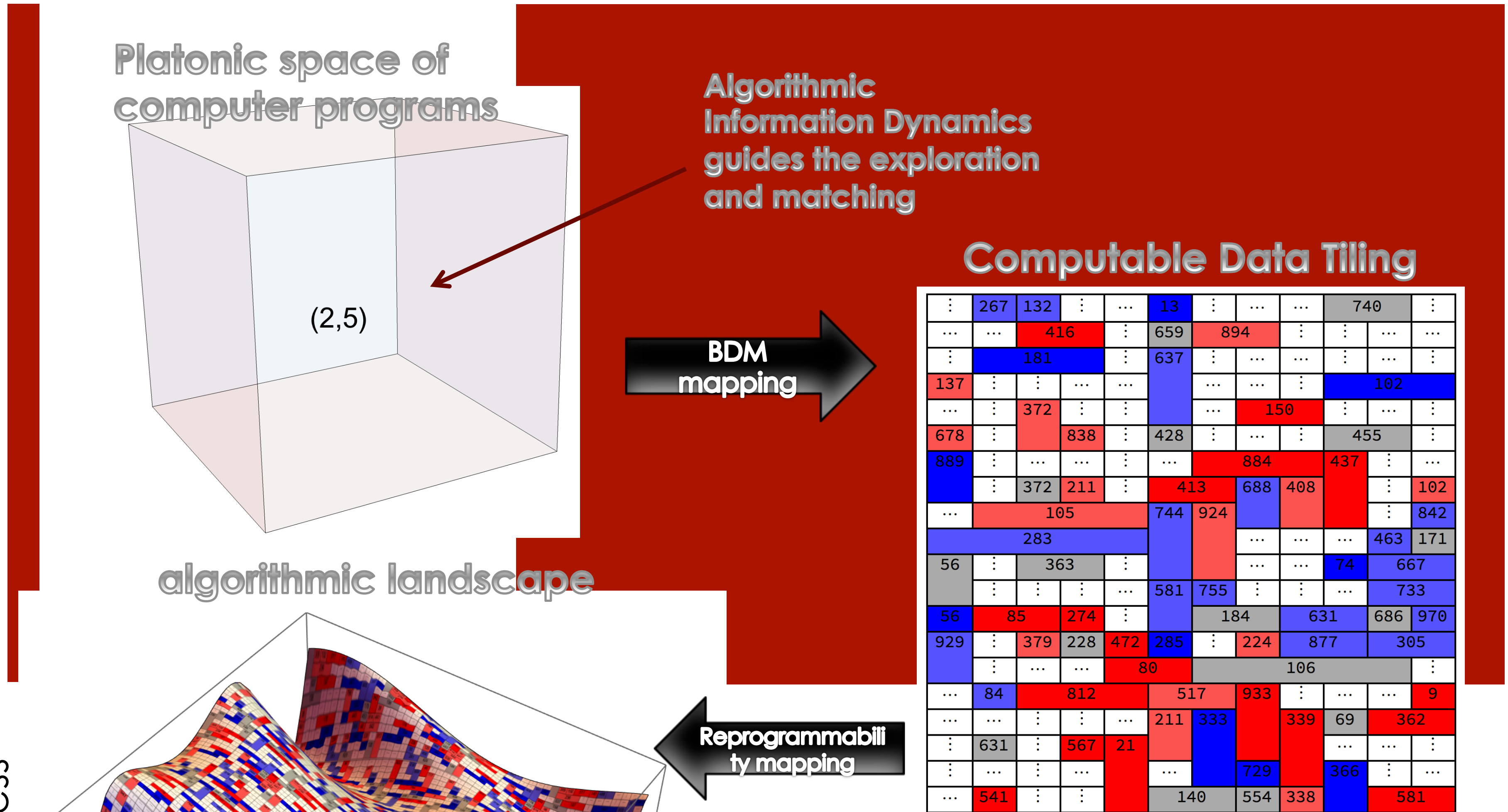
H Zenil, et al.
A Decomposition Method for Global Evaluation of Shannon Entropy and Local Estimations of Algorithmic Complexity, Entropy, 20(8), 605, 2018.

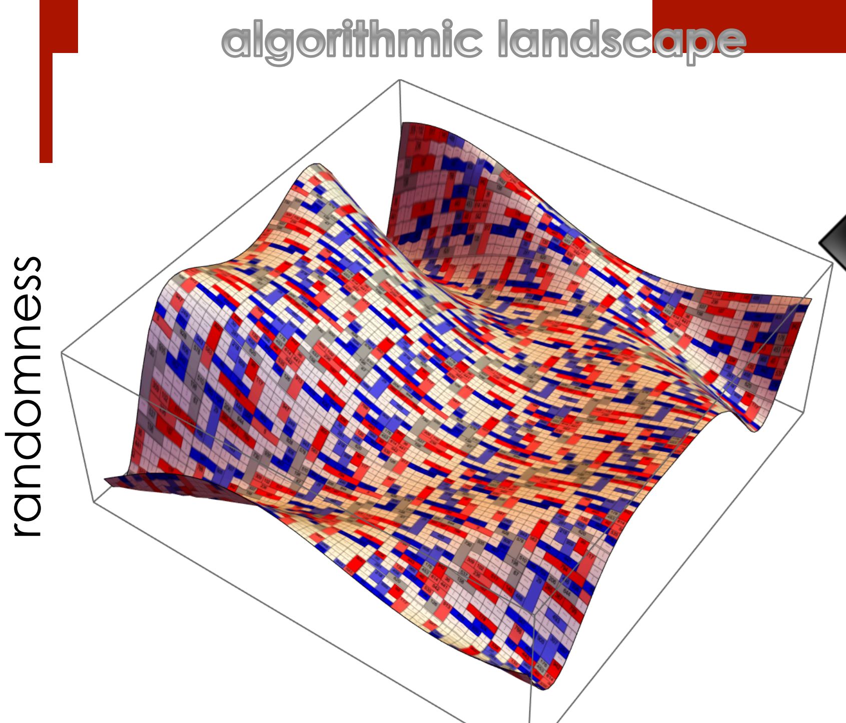
Block Decomposition Method



- No overlapping: Under-estimation
- Overlapping: Over-estimation

H Zenil, et al.
A Decomposition Method for Global Evaluation of Shannon Entropy and Local Estimations of Algorithmic Complexity, Entropy, 20(8), 605, 2018.





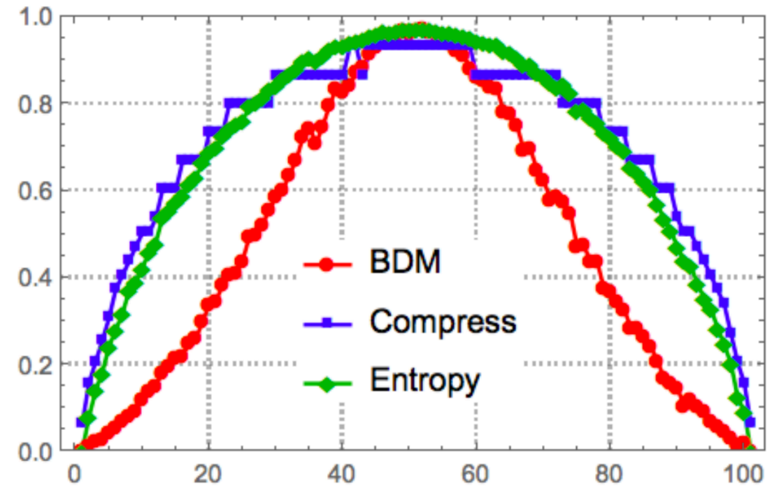
H Zenil, et al.
A Decomposition Method for Global Evaluation of Shannon Entropy and Local Estimations of Algorithmic Complexity, Entropy, 20(8), 605, 2018.

Block Decomposition Method (BDM)

$$D(n,k)(x) = \frac{|\{T \in (n,k) : T \textit{ produces } x\}|}{|\{T \in (n,k) : T \textit{ halts }\}|}$$

$$CTM(x,n,k) = -\log_b(D(n,k)(x))$$

$$BDM(X,l,m) = \sum_i^k CTM(x_i,n,k) + \log(s_i)$$

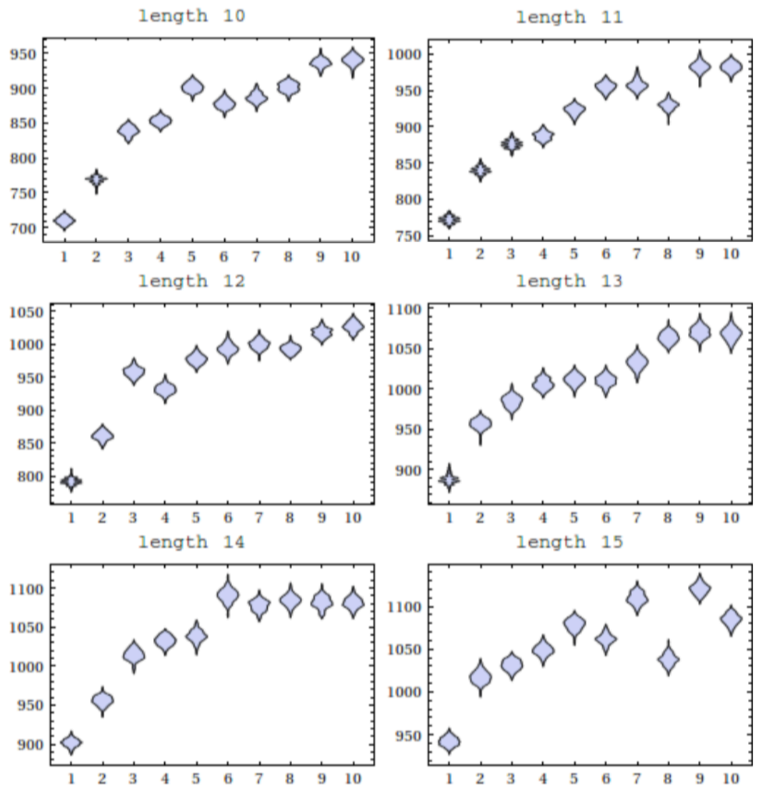


E.g. These two strings are of lower randomness than what Entropy would suggest:
011011010010, 010101110100

H Zenil, et al.
A Decomposition Method for Global Evaluation of Shannon Entropy and Local Estimations of Algorithmic Complexity, Entropy, 20(8), 605, 2018.

BDM v Lossless Compression

The transition between one method and the other. What is complex for the Coding Theorem method is also less compressible.



All 2^n bit strings for small n sorted by CTM versus lossless compression

[F. Soler-Toscano, H. Zenil et al. *Computability* (2013)]

Correlation with number of instructions (K_m = CTM)

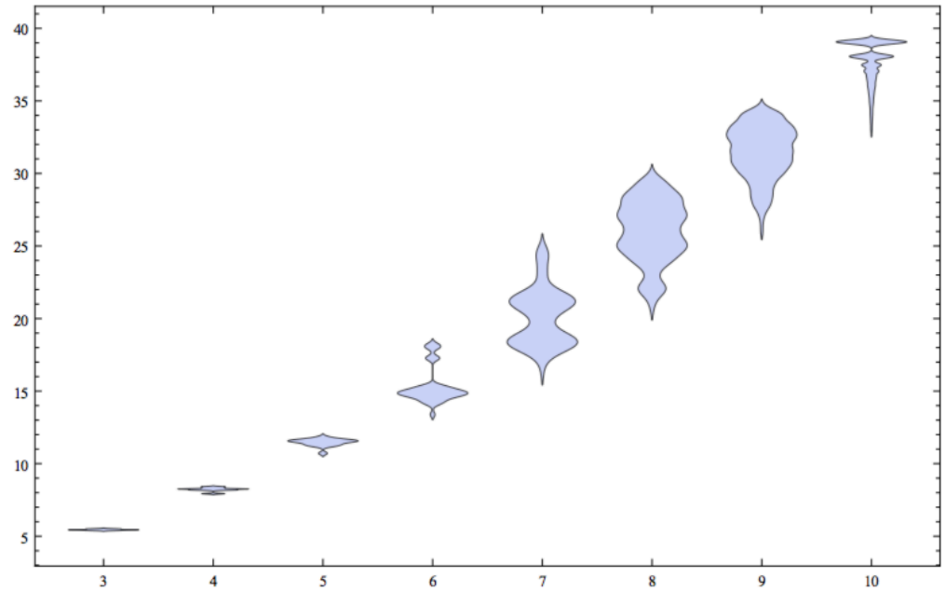
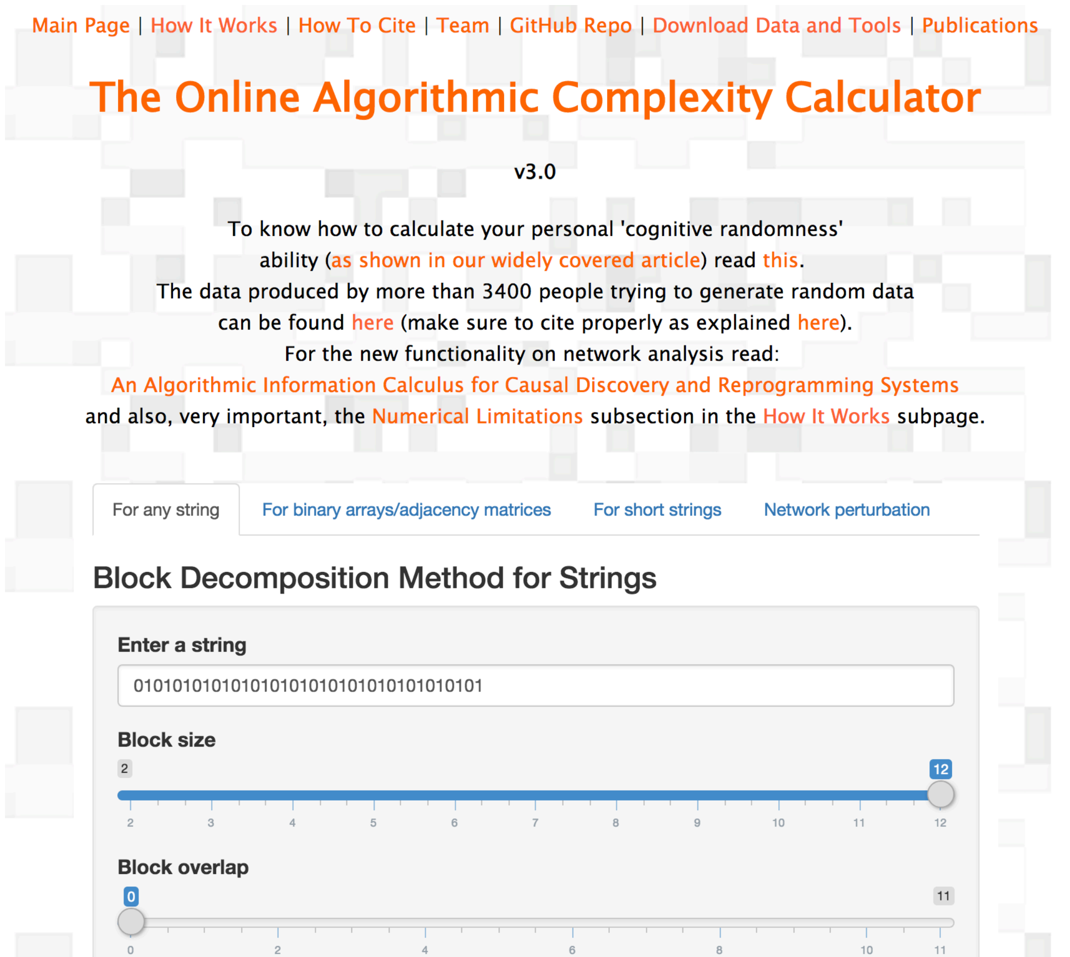


Figure 1: Distribution chart of K_m values according to the minimum number of instructions required. Each "drop-like" distribution is the set of strings that are minimally produced with the same number of instructions (horizontal axis). The more instructions needed to produce the strings, the more complex they are (vertical axis in K_m units).

[H. Zenil and J-P. Delahaye, *Computability*; 2013]

The Online Algorithmic Complexity Calculator



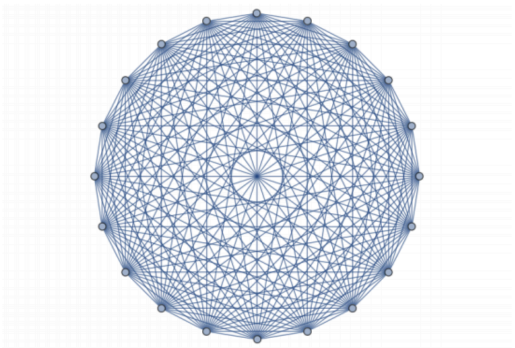
complexitycalculator.com

Graphs and Networks

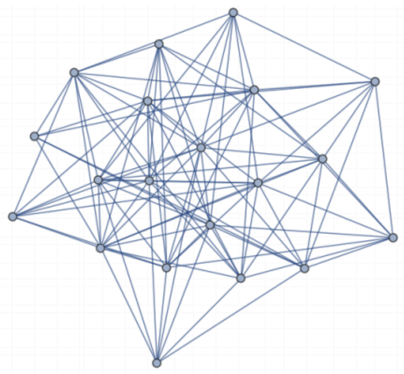
Graph complexity

M. Gell-Mann (Nobel Prize 1969) thought that any reasonable measure of complexity of graphs should have both completely disconnected and completely connected graphs to have minimal complexity (*The quark and the jaguar*, 1994).

Unlike Graph Entropy, Graph Kolmogorov complexity is robust:



complete graph: $K \sim \log(|N|)$



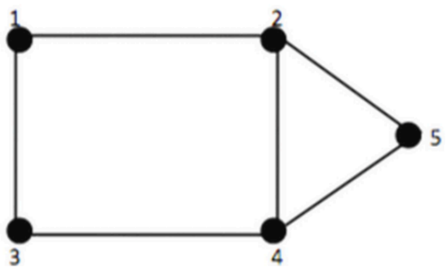
E-R random graph: $K \sim |E|$

Graph Kolmogorov complexity

Complete and disconnected graphs with $|N|$ nodes have low (algorithmic) information content. In a random graph every edge $e \in E$ requires some information to be described. Both $K(G) \sim K(Adj(G))$!

Numerical estimations to $K(G)$

What is the Kolmogorov complexity of an adjacency matrix?



$$A_d = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

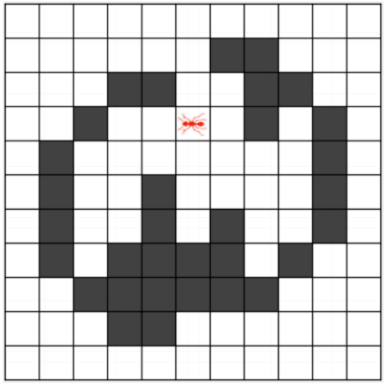
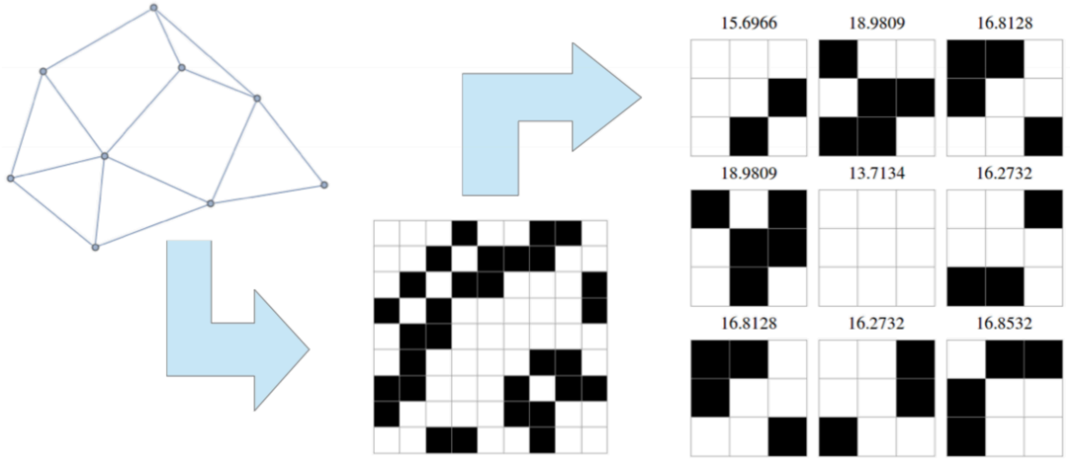


Figure: Two-dimensional Turing machines, also known as Turmites (Langton, Physica D, 1986). This idea can be generalized to *n*-dimensional 'tapes'.

[Zenil et al. Physica A, 2014]

Graph algorithmic probability

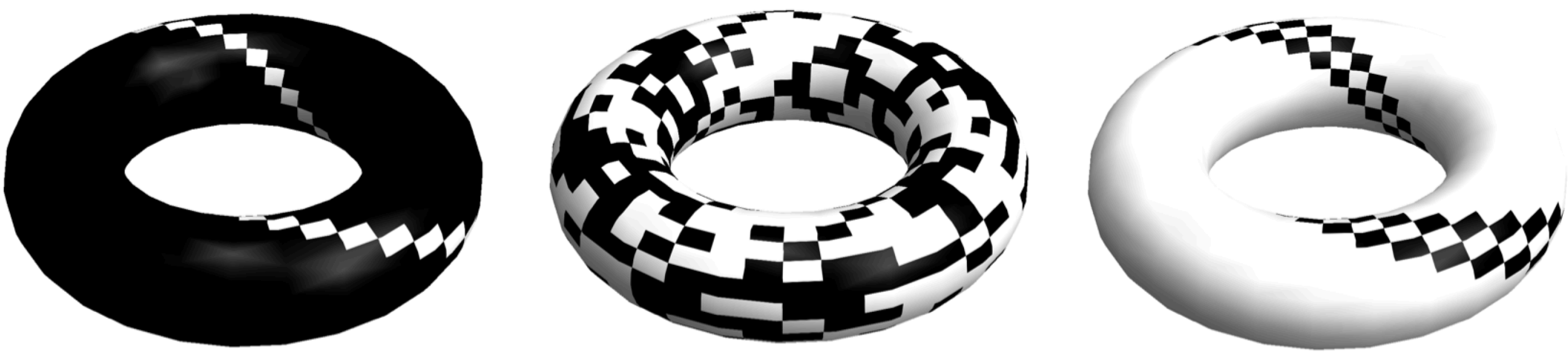


- Labelled complexity is a good approximation of unlabelled.
- Different boundary conditions provide a solution to the boundaries problem (cyclic, overlapping).
- Overlapping sub matrices avoids the problem of permuting squares with same complexity (leads to overfitting).
- The best option is to recursively divide into square matrices for which exact complexity estimations are known.
- Numerically sound and robust.

[Zenil et al. Physica A: Statistical Mechanics and Its Applications (2014)]

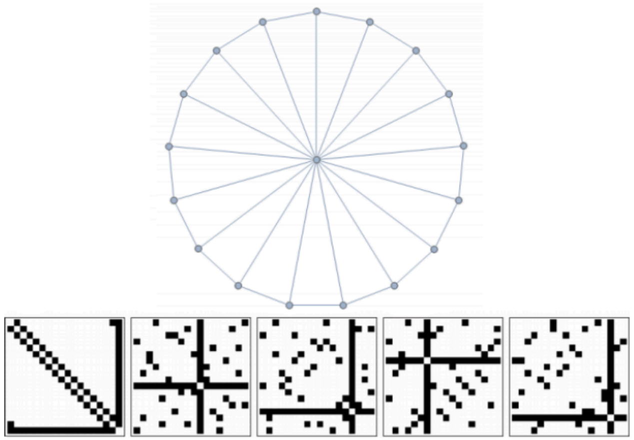
Boundary Conditions

$$BDM(X, l, m) = \sum_{i}^{k} CTM(x_i, n, k) + \log(s_i)$$



H Zenil, F Soler-Toscano, N.A. Kiani, S. Hernández-Orozco, A. Rueda-Toicen
A Decomposition Method for Global Evaluation of Shannon Entropy and Local Estimations of Algorithmic Complexity, Entropy 20(8), 605, 2018.

BDM and graph automorphism group



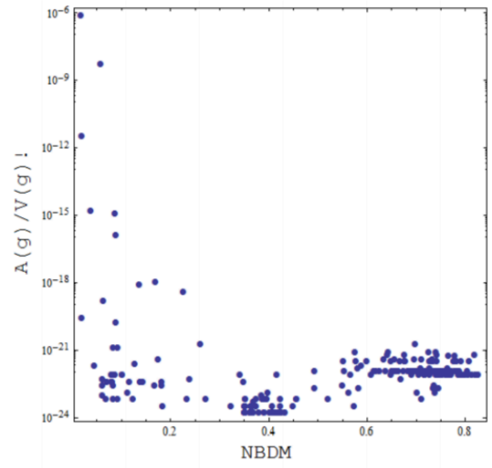


Figure: Left: An adjacency matrix is not a unlabelled graph invariant yet isomorphic graphs have similar K. Right: Graphs with large automorphism group size (group symmetry) have lower K.

This correlation shows that for low algebraic complexity of labelled graphs (large automorphism count) as measure on a single adjacency matrix, any labelled graph is a good approximation of the algorithmic complexity of the graph isomorphism group. is captured by the complexity of their adjacency matrix (which is a labelled graph object).

[Zenil et al. Physica A (2014)]

Unlabelled Graph Complexity

Graph unlabelled Kolmogorov complexity:

Definition

Graph Unlabelled Kolmogorov Complexity: Let $Adj(G)$ be the adjacency matrix of G and $Aut(G)$ its automorphism group, then,

$$K(G) = \min\{K(Adj(G)) | Adj(G) \in A(Aut(G))\}$$

where $A(Aut(G))$ is the set of adjacency matrices of all $G \in Aut(G)$. (The problem is believed to be in NP but not in NP-complete).

Labelled graph complexity = unlabelled graph complexity up to a constant c. Proof sketch: There is an algorithm (e.g. brute force) of finite (small) size c that produces any isomorphic graph from any other (even if in **NP**).

[Zenil, Kiani and Tegnér (Seminars in Cell and Developmental Biology), 2016]

Graph tests

Definition

Dual graph: A dual graph of a plane graph G is a graph that has a vertex corresponding to each face of G, and an edge joining two neighboring faces for each edge in G.

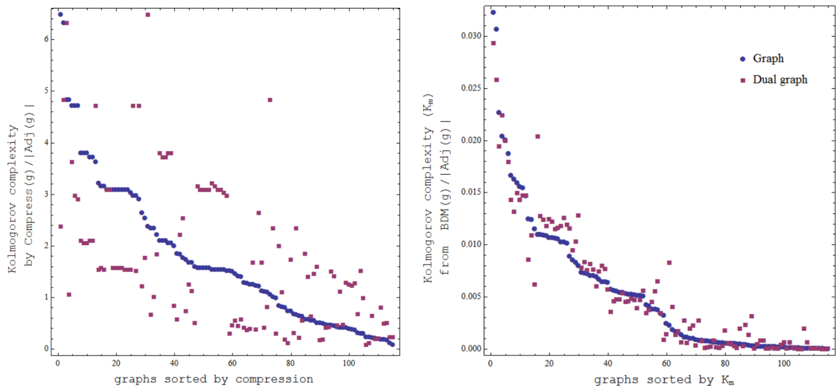
Definition

Graph spectra: The set of graph eigenvalues of the adjacency matrix is called the spectrum of the graph. The Laplacian matrix of a graph is sometimes also known as the graph's spectrum.

Definition

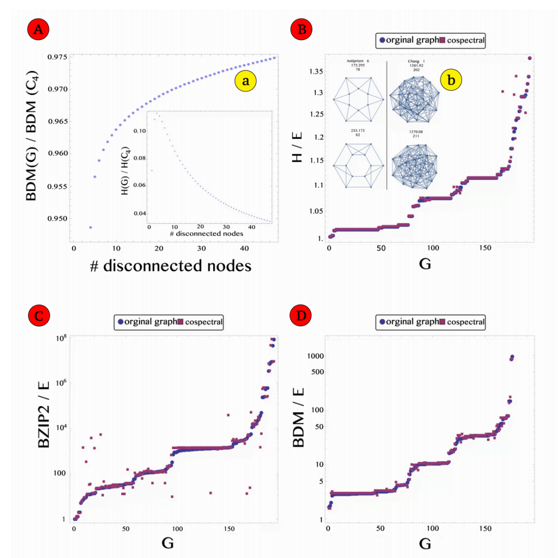
Cospectral graphs: Two graphs are called *isospectral* or *cospectral* if they have the same spectra.

Testing compression and BDM on dual graphs



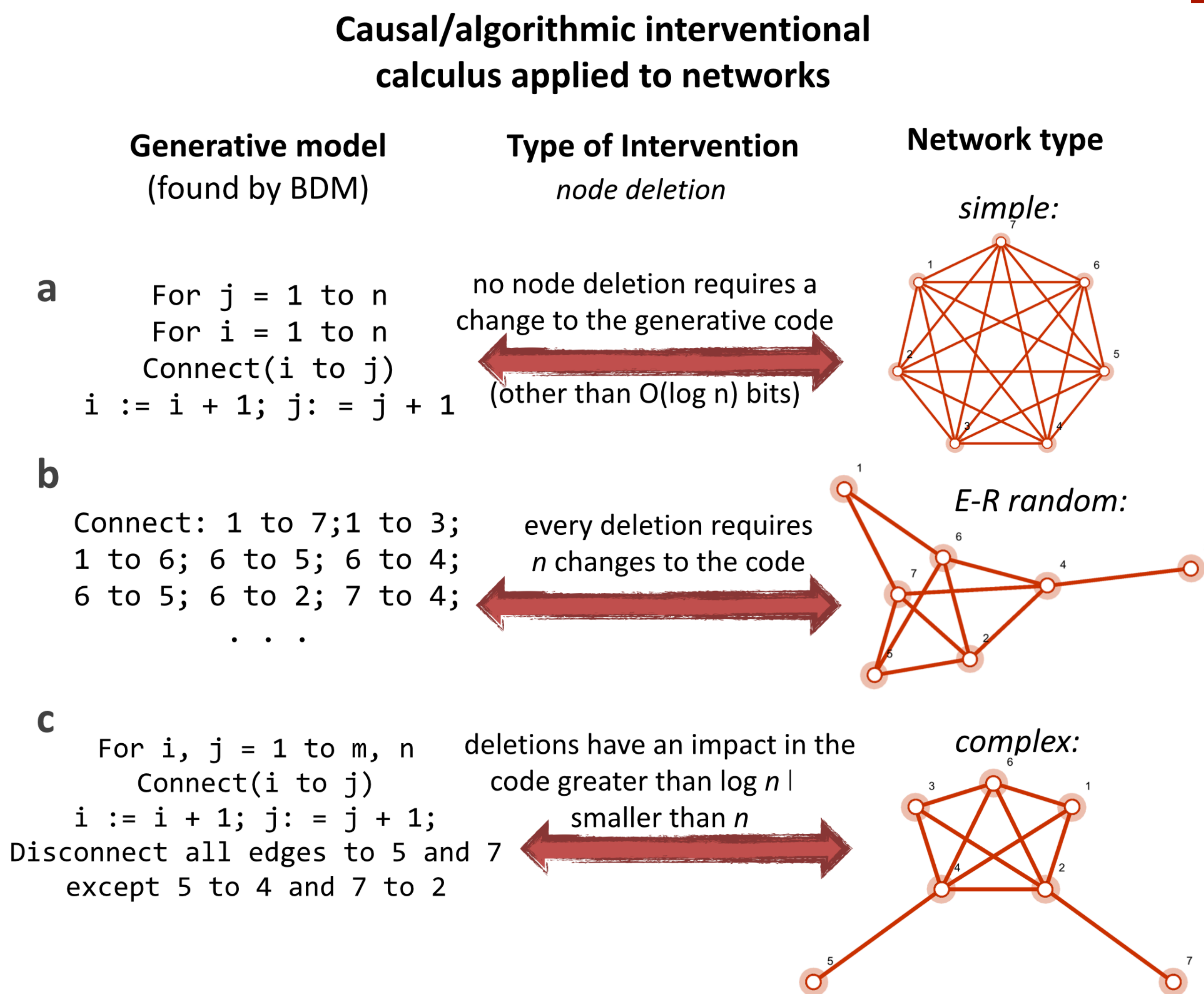
[Zenil et al. Physica A (2014)]

H, compression and BDM on cospectral graphs



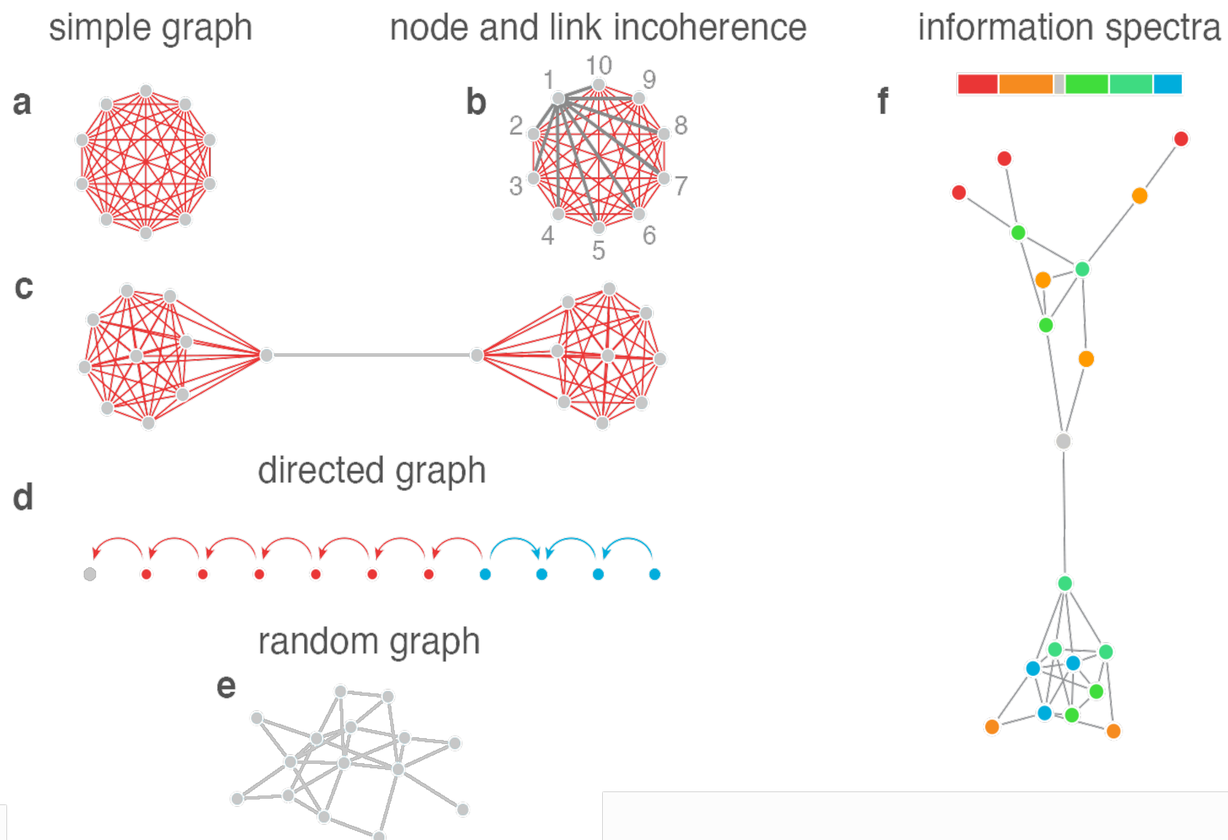
Algorithmic Information Dynamics

Perturbation analysis



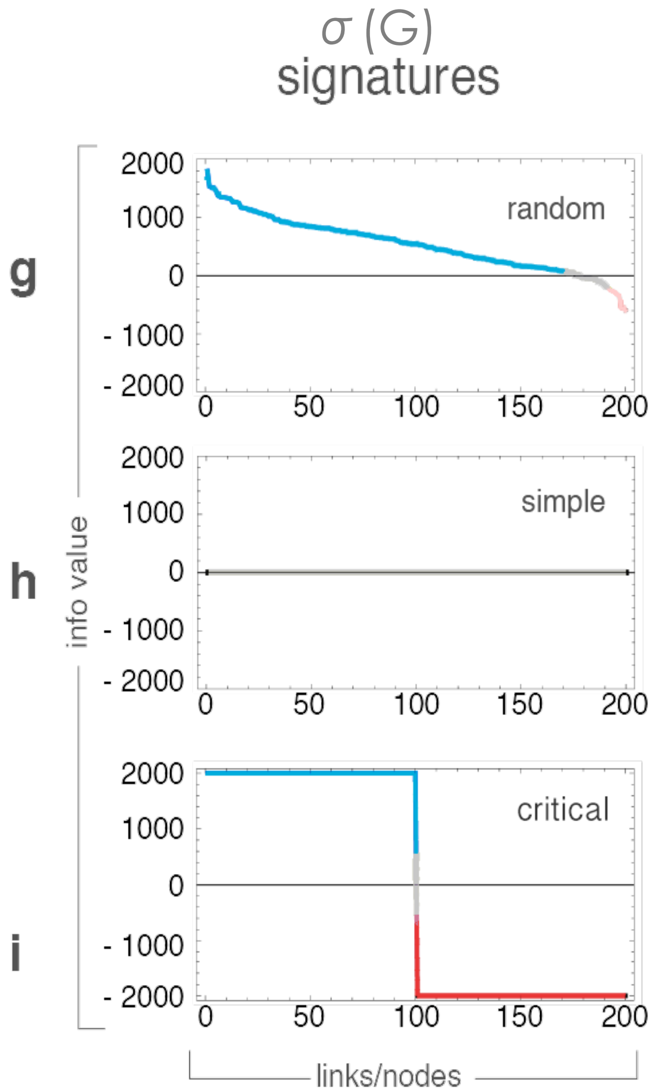
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér,
An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems, bioaRXiv DOI: https://doi.org/10.1101/185637

Algorithmic Information Dynamics: A Calculus of Algorithmic Information Change



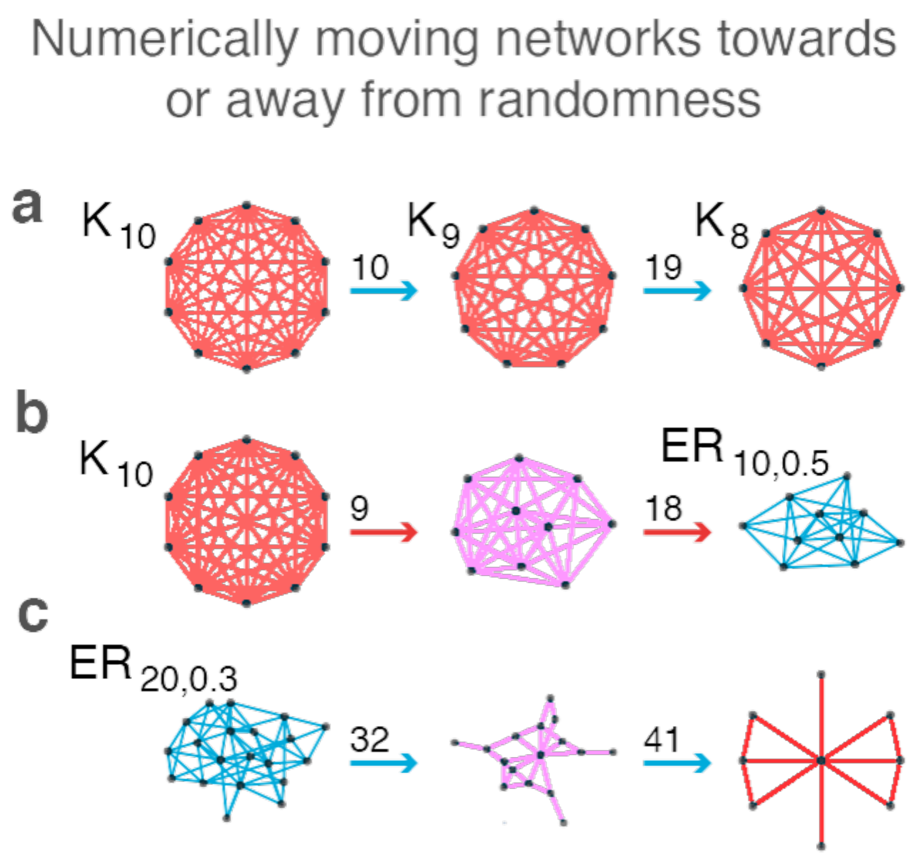
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér, An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems, bioaRXiv DOI: https://doi.org/10.1101/185637

Information Signatures



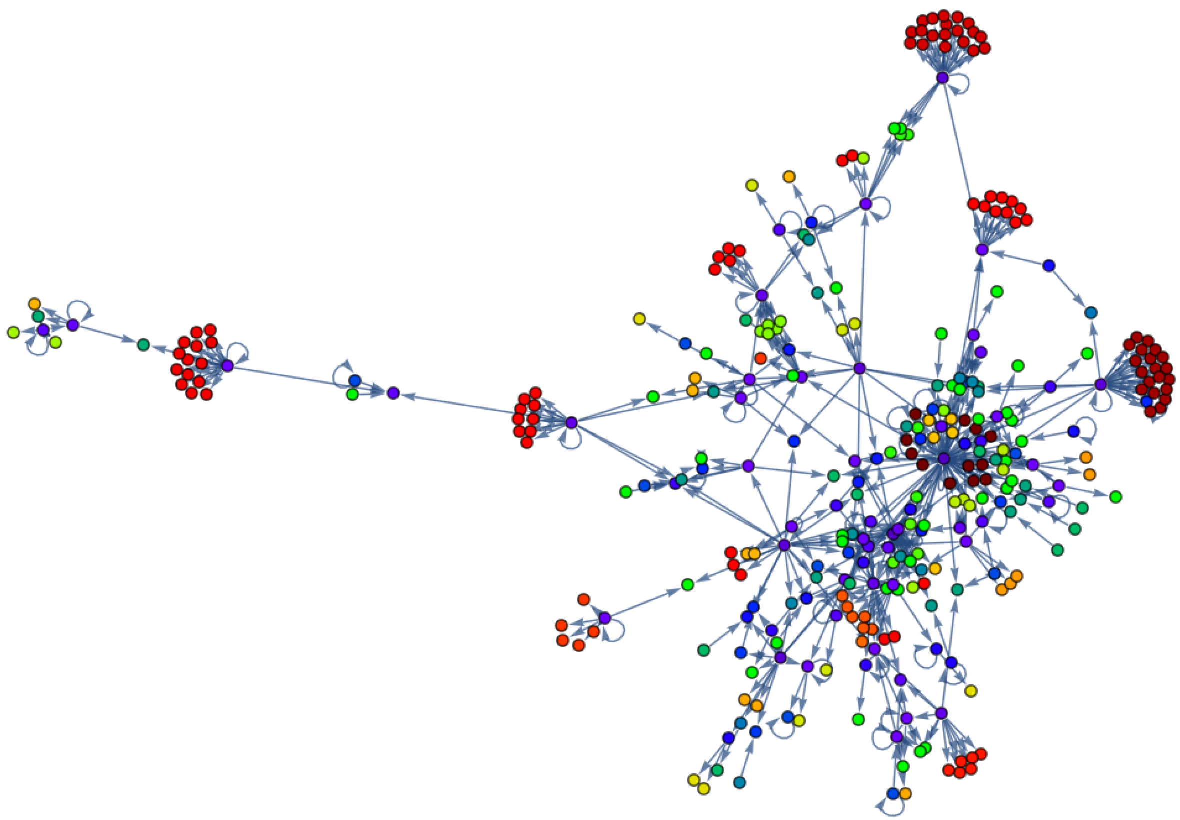
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér, An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems, bioaRXiv DOI: https://doi.org/10.1101/185637

Moving Networks



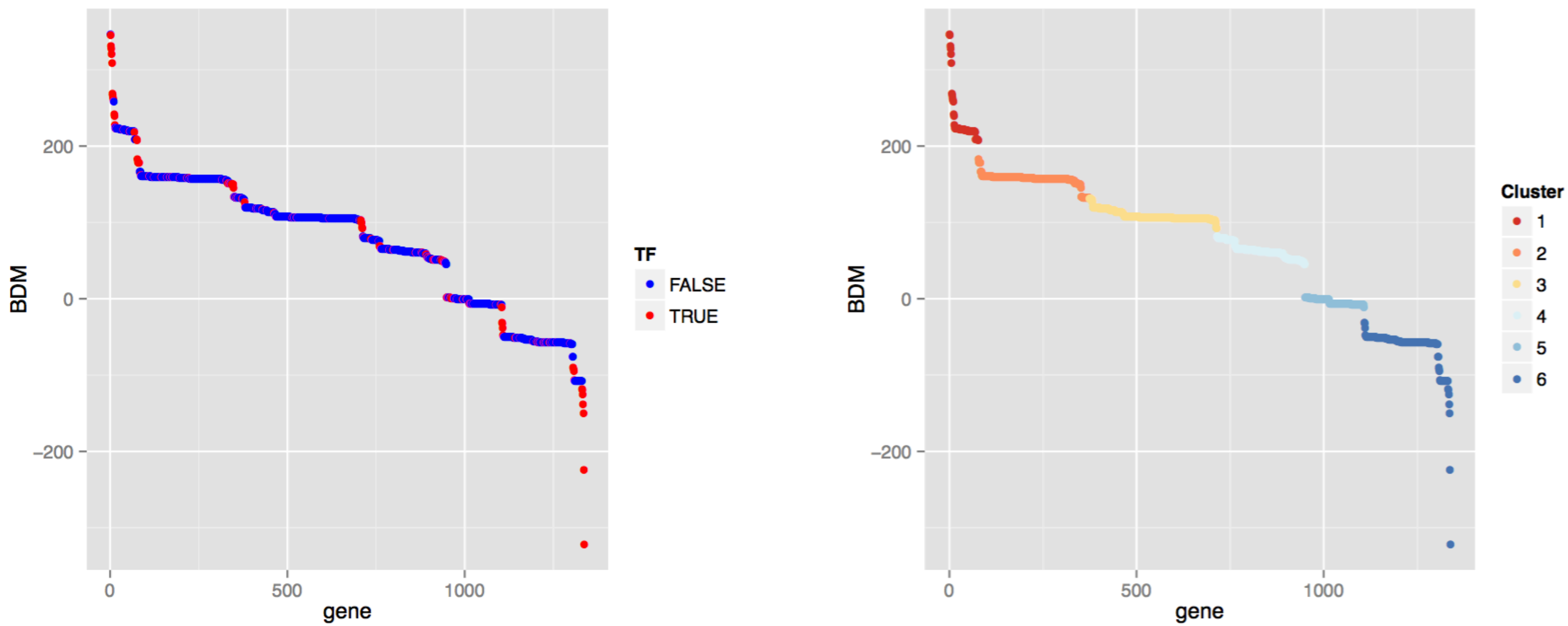
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér,
An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems, bioaRXiv DOI: https://doi.org/10.1101/185637

E.Coli experimentally validated TF-network



An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Block Decomposition Method

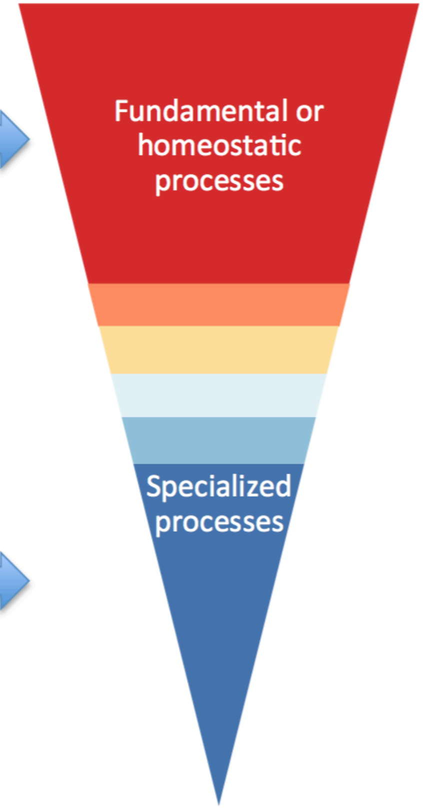


Six clusters were selected, using partitioning around medoids clustering. The number of clusters was estimated by optimum average silhouette width

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Block Decomposition Method

	GO.ID	Term	Pval
Cluster 1	GO:0006094	gluconeogenesis	1.60E-06
	GO:0006096	glycolysis	0.00036
	GO:0008615	pyridoxine biosynthetic process	0.0124
	GO:0009255	Entner-Doudoroff pathway	0.0124
	GO:0042330	taxis	0.02035
	GO:0016052	carbohydrate catabolic process	0.02911
2	-	-	-
3	-	-	-
4	-	-	-
5	-	-	-
Cluster 6	GO:0006793	phosphorus metabolic process	2.10E-08
	GO:0009252	peptidoglycan biosynthetic process	2.90E-07
	GO:0006777	Mo-molybdopterin cofactor biosynthetic process	1.20E-05
	GO:0009086	methionine biosynthetic process	0.0027
	GO:0009242	colanic acid biosynthetic process	0.0124
	GO:0006164	purine nucleotide biosynthetic process	0.0196
	GO:0009228	thiamine biosynthetic process	0.0254
	GO:0009243	O antigen biosynthetic process	0.0254

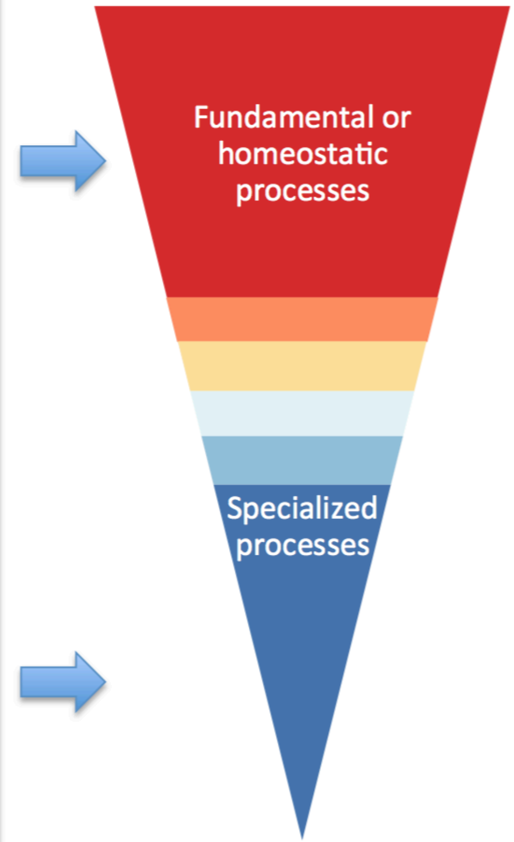


Gene Ontology (**Biological Process**):
over-represented categories tested with TopGO weight01 method (Fisher p<0.05)

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Block Decomposition Method

	KEGG ID	Term	Pval
Cluster 1	00010	Glycolysis / Gluconeogenesis	1.76E-08
	00051	Fructose and mannose metabolism	7.13E-06
	02030	Bacterial chemotaxis	6.32E-05
	02020	Two-component system	7.55E-04
	00620	Pyruvate metabolism	4.08E-03
	00030	Pentose phosphate pathway	5.14E-03
	02060	Phosphotransferase system (PTS)	5.45E-03
	00680	Methane metabolism	6.70E-03
	01110	Biosynthesis of secondary metabolites	9.59E-03
	01120	Microbial metabolism in diverse environments	1.44E-02
2	-	-	-
3			
4	-	-	-
5	-	-	-
Cluster 6	00550	Peptidoglycan biosynthesis	1.01E-07
	01100	Metabolic pathways	6.74E-04
	04122	Sulfur relay system	4.11E-03
	00621	Dioxin degradation	9.20E-03
	00622	Xylene degradation	9.20E-03
	00360	Phenylalanine metabolism	1.48E-02
	00300	Lysine biosynthesis	2.48E-02
	00230	Purine metabolism	3.50E-02
	00670	One carbon pool by folate	3.73E-02

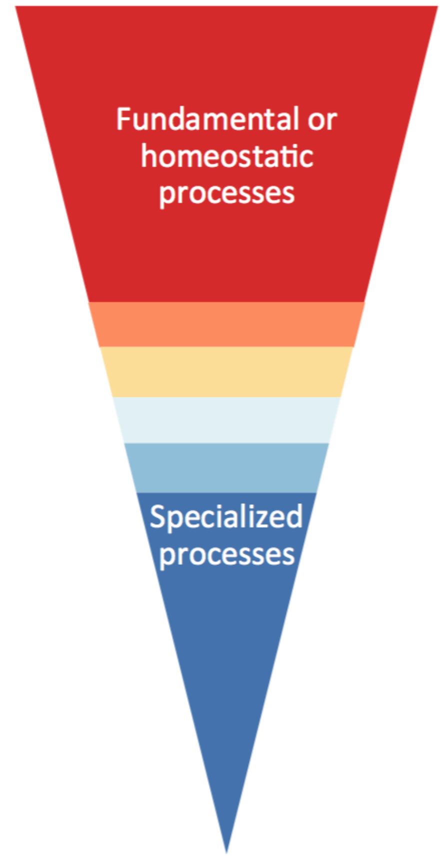


Over-represented **KEGG pathways** (p<0.05)

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

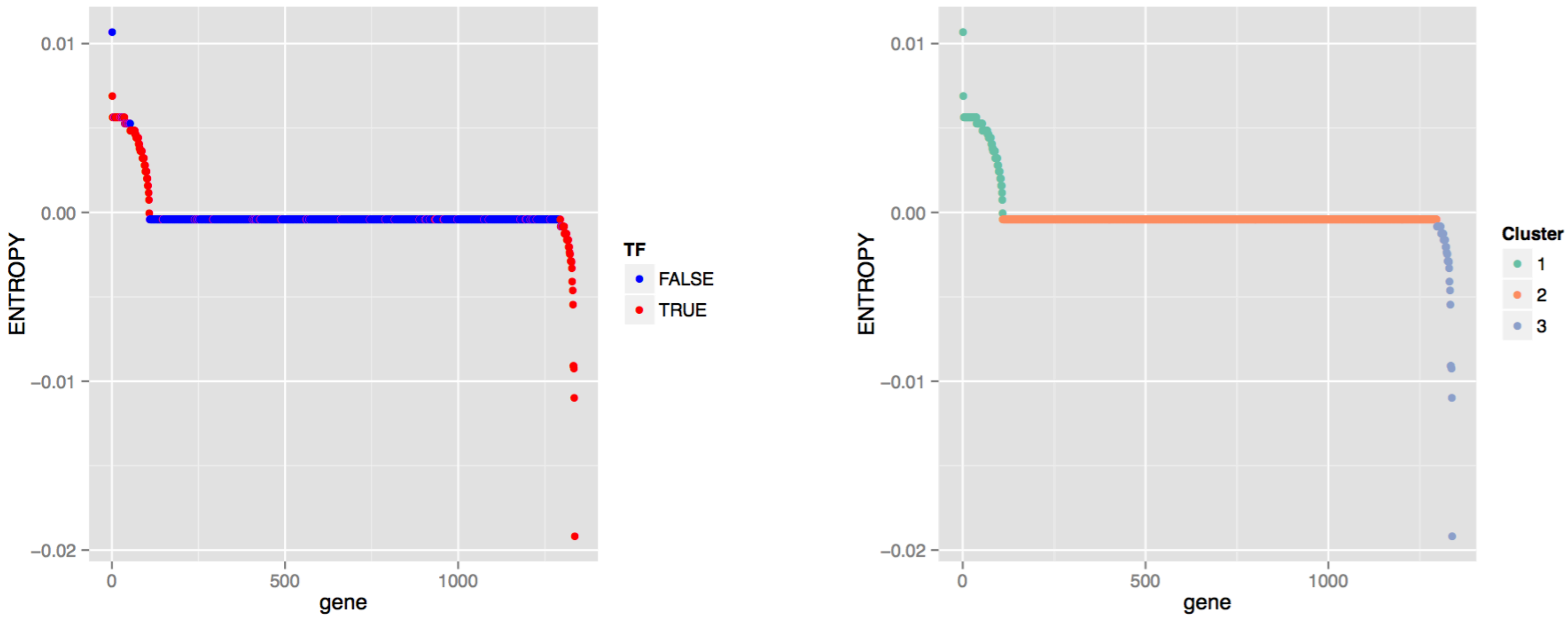
	EcoCyc pathway	Term
Cluster 1	superpathway of glycolysis and Entner-Doudoroff	5.37E-07
	Sugar Alcohols Degradation	4.82E-06
	superpathway of hexitol degradation (bacteria)	1.91E-05
	glycolysis I (from glucose-6P)	1.91E-05
	glycolysis II (from fructose-6P)	1.91E-05
	gluconeogenesis I	2.56E-04
	Gluconeogenesis	2.56E-04
	Sugar Derivatives Degradation	0.003115401
	Secondary Metabolites Degradation	0.003131693
	superpathway of glycolysis, pyruvate dehydrogenase, TCA, and glyoxylate bypass	0.004830985
	TCA cycle	0.004830985
	Glycolysis	0.005196795
	Generation of Precursor Metabolites and Energy	0.005701038
	sedoheptulose bisphosphate bypass	0.037381258
	Entner-Duodoroff Pathways	0.037381258
	Entner-Doudoroff pathway I	0.037381258
	CpxAR Two-Component Signal Transduction System	0.037381258
	Signal transduction pathways	0.045972995
2	-	-
3	-	-
4	-	-
5	-	-
Cluster 6	methylphosphonate degradation I	9.40E-06
	Phosphorus Compounds Metabolism	9.40E-06
	Methylphosphonate Degradation	9.40E-06
	Pyrimidine Nucleobases Degradation	0.003167986
	Uracil Degradation	0.003167986
	uracil degradation III	0.003167986
	peptidoglycan biosynthesis (meso-diaminopimelate containing)	0.003167986
	Peptidoglycan Biosynthesis	0.003167986
	Cell Wall Biosynthesis	0.003167986
	putrescine degradation II	0.005063846
	3-phenylpropionate and 3-(3-hydroxyphenyl)propionate degradation	0.018877832
	proline to cytochrome bo oxidase electron transfer	0.019695489
	UDP-N-acetylmuramoyl-pentapeptide biosynthesis I (meso-DAP-containing)	0.028546946
	UDP-N-Acetylmuramoyl-Pentapeptide Biosynthesis	0.028546946
	2-oxopentenoate degradation	0.04015748
	Putrescine Degradation	0.0413727
	Pyrimidine Nucleotides Degradation	0.06959294
	superpathway of ornithine degradation	0.075477235
	Purine Nucleotides De Novo Biosynthesis	0.075477235
	superpathway of purine nucleotides de novo biosynthesis II	0.075477235
	superpathway of arginine, putrescine, and 4-aminobutyrate degradation	0.09681385
	L-rhamnose degradation I	0.09815362
	L-rhamnose Degradation	0.09815362

Over-represented **EcoCyc pathways** (FDR<0.05)



An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

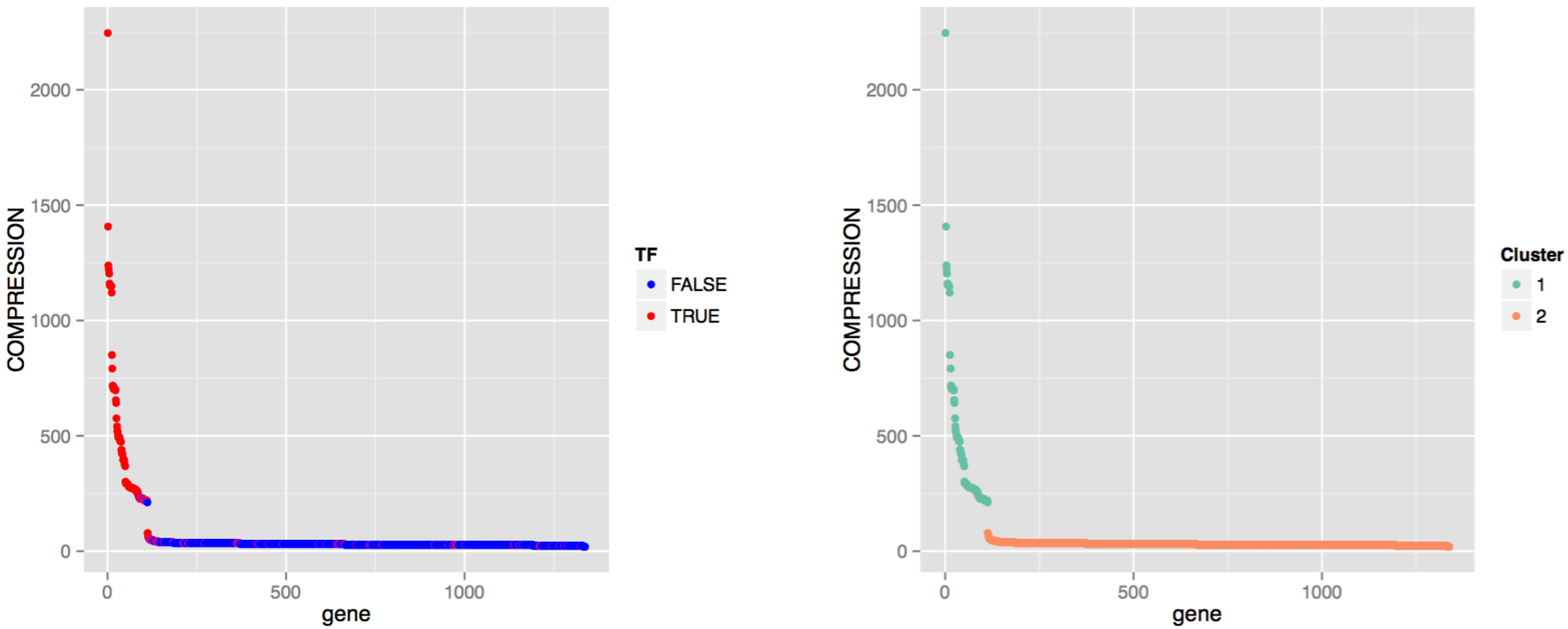
Entropy



Three clusters were identified (above baseline, baseline, below baseline).
Non-baseline nodes are enriched for Transcription Factors

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Compression



Two clusters were identified (above baseline, baseline).
Above-baseline nodes are enriched for Transcription Factors

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

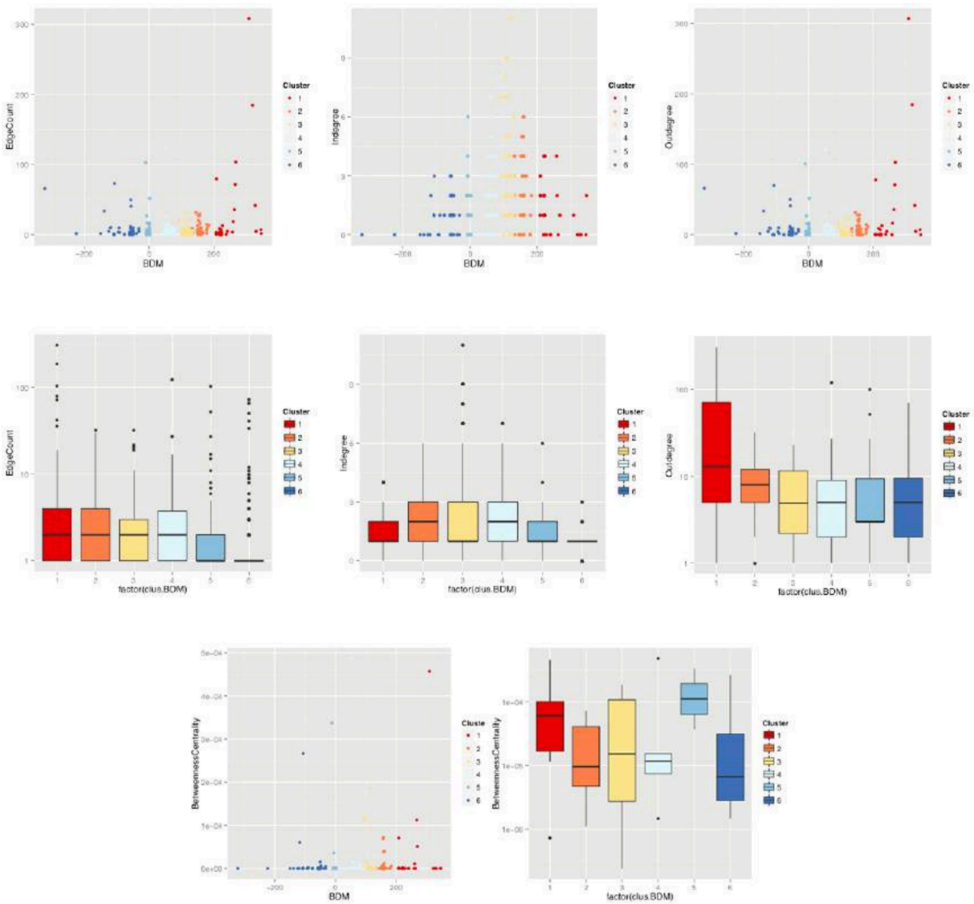
Non significant clusters using Compression

	GO.ID	Term	Pval
Cluster 1	GO:0006805	xenobiotic metabolic process	0.003
	GO:0009255	Entner-Doudoroff pathway	0.014
	GO:0006355	**regulation of transcription**, DNA-dependent	0.029
Cluster 2	-	-	

Gene Ontology (Biological Process): Over-represented categories tested with TopGO weight01 method (Fisher $p<0.05$) using lossless compression (Compress algorithm).

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

BDM sensitivity and specificity



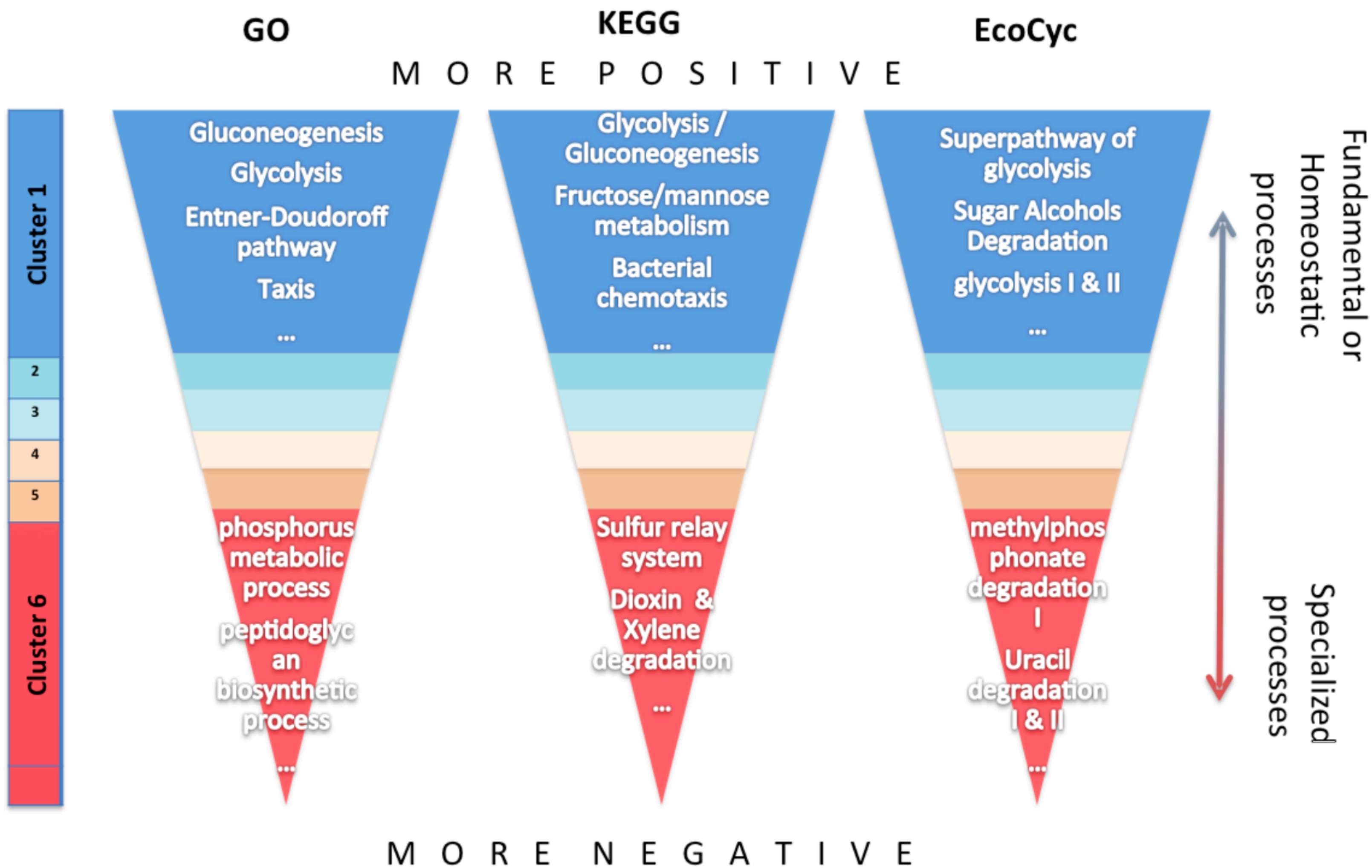
BDM did not correlate with any trivial graph-theoretic measure such as:

- Node degree
- In degree
- Out degree
- Betweeness Centrality
- Entropy
- Compression

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

E.coli TF network

Info nodes of the e.coli, the most studied organism and genetic network:



An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Resource to check: Fundamental → Conserved

> These findings point at an emerging picture in which a core of enzyme activities involving amino acid, energy, carbohydrate and lipid metabolism have evolved to provide the basic functions required for life. However, as indicated by the relatively low number of significantly modular links, the precise complement of enzymes associated within this core for each species is flexible. It is important to remember that the network

Open Access

Research

The conservation and evolutionary modularity of metabolism

José M Peregrín-Alvarez*†, Chris Sanford*‡ and John Parkinson*‡§

Addresses: *Program in Molecular Structure and Function, Hospital for Sick Children, College Street, Toronto, ON, M5G 1L7, Canada. †Department of Molecular Biology and Biochemistry, University of Malaga, Avda. Cevantes, 29071 Malaga, Spain. ‡Department of Molecular Genetics, University of Toronto, King's College Circle, Toronto, ON, M5S 1A8, Canada. §Department of Biochemistry, University of Toronto, Toronto, Kings' College Circle, ON, M5S 1A8, Canada.

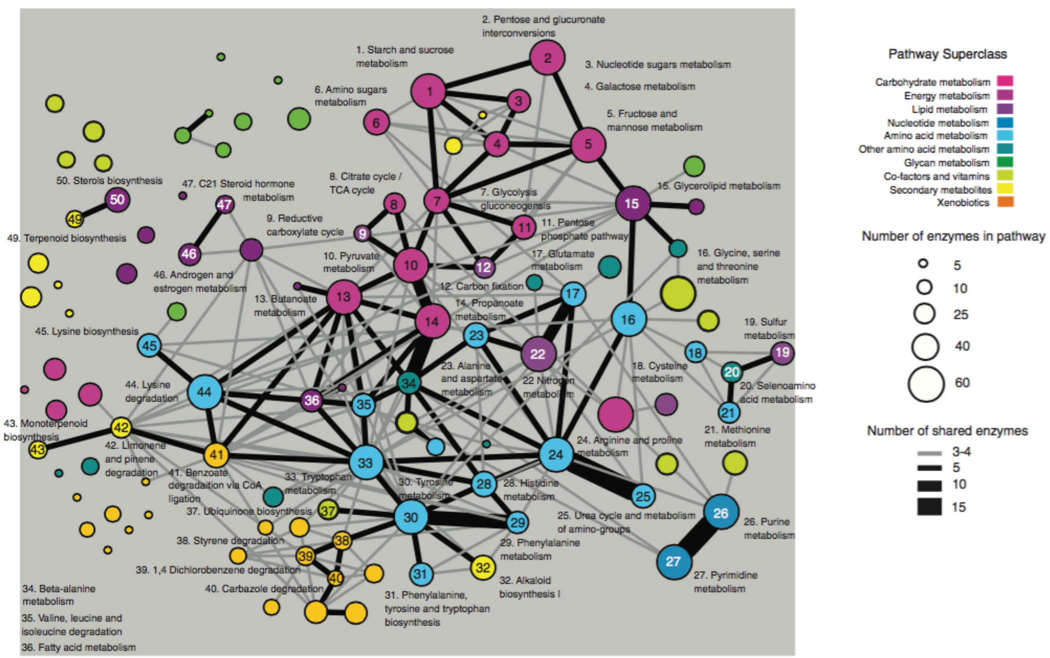
Correspondence: José M Peregrín-Alvarez. Email: peregrin@sickkids.ca. John Parkinson. Email: jparkin@sickkids.ca

Published: 12 June 2009

Genome **Biology** 2009, **10:**R63 (doi:10.1186/gb-2009-10-6-r63)

The electronic version of this article is the complete one and can be found online at http://genomebiology.com/2009/10/6/R63

Received: 5 February 2009
Revised: 27 May 2009
Accepted: 12 June 2009



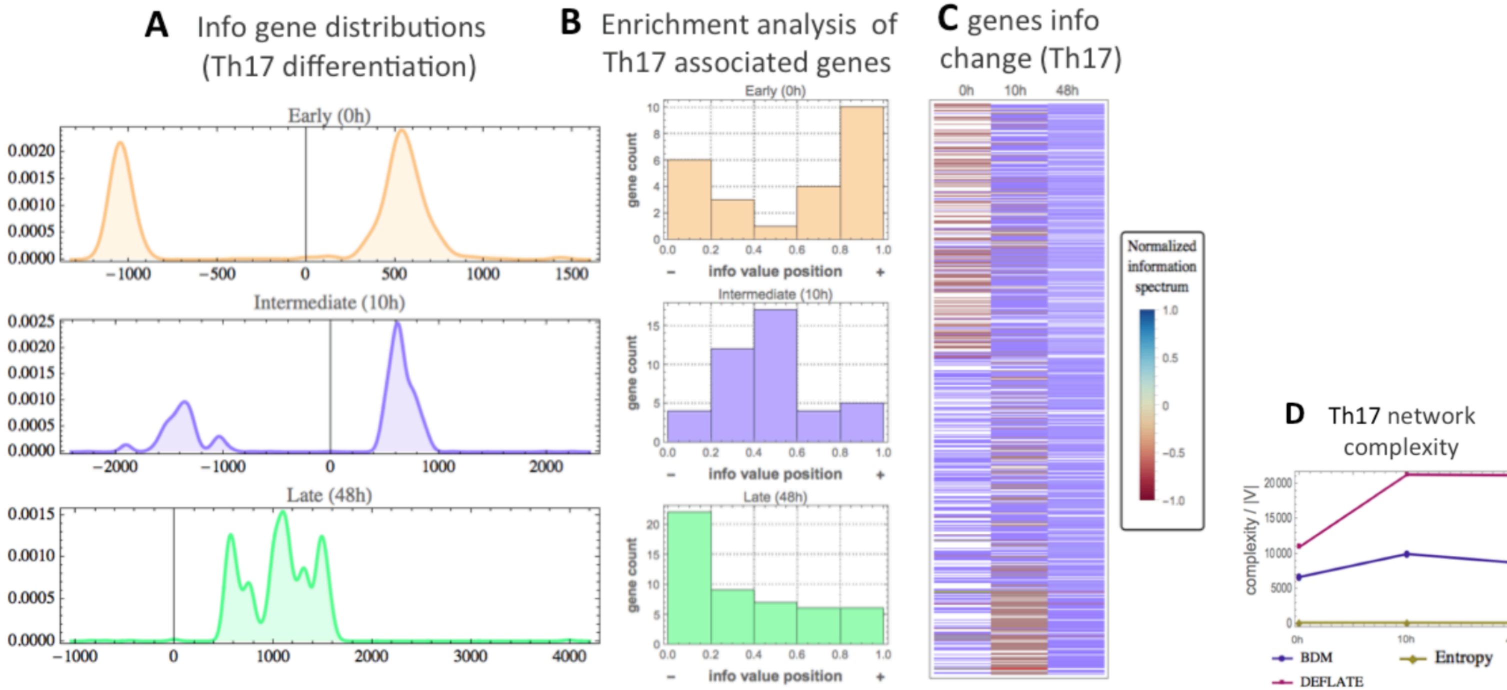
Abstract

Background: Cellular metabolism is a fundamental biological system consisting of myriads of enzymatic reactions that together fulfill the basic requirements of life. The recent availability of vast amounts of sequence data from diverse sets of organisms provides an opportunity to systematically examine metabolism from a comparative perspective. Here we supplement existing genome and protein resources with partial genome datasets derived from 193 eukaryotes to present a comprehensive survey of the conservation of metabolism across 26 taxa representing the three domains of life.

Results: In general, metabolic enzymes are highly conserved. However, organizing these enzymes within the context of functional pathways revealed a spectrum of conservation from those that are highly conserved (for example, carbohydrate, energy, amino acid and nucleotide metabolism enzymes) to those specific to individual taxa (for example, those involved in glycan metabolism and secondary metabolite pathways). Applying a novel co-conservation analysis, KEGG defined pathways did not generally display evolutionary coherence. Instead, such modularity appears restricted to smaller subsets of enzymes. Expanding analyses to a global metabolic network revealed a highly conserved, but nonetheless flexible, 'core' of enzymes largely involved in multiple reactions across different pathways. Enzymes and pathways associated with the periphery of this network were less well conserved and associated with taxon-specific innovations.

Conclusions: These findings point to an emerging picture in which a core of enzyme activities involving amino acid, energy, carbohydrate and lipid metabolism have evolved to provide the basic functions required for life. However, the precise complement of enzymes associated within this core for each species is flexible.

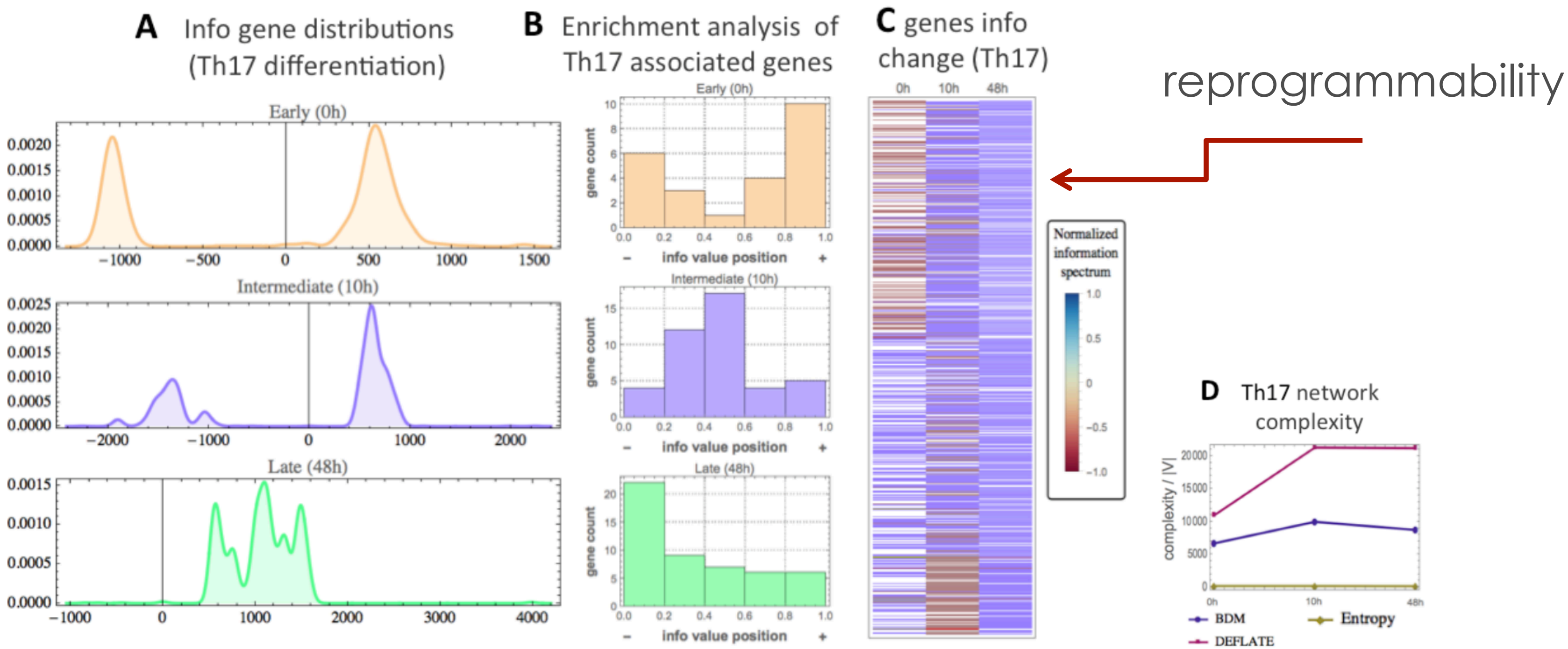
Information dynamics of biological cells



Info nodes of a T_h17 mouse cell in its differentiation process (gene expression, siRNA). A: Distribution of info genes. B: Non-uniform distribution of T_h17 marker genes. C: Information change in the 3 time steps. D: Overal network complexity.

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

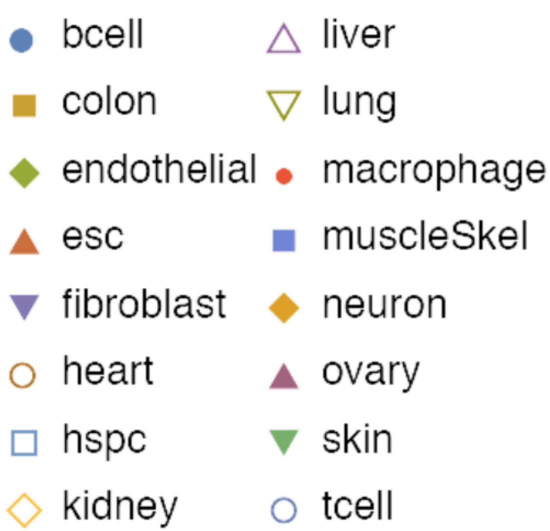
Information dynamics of biological cells

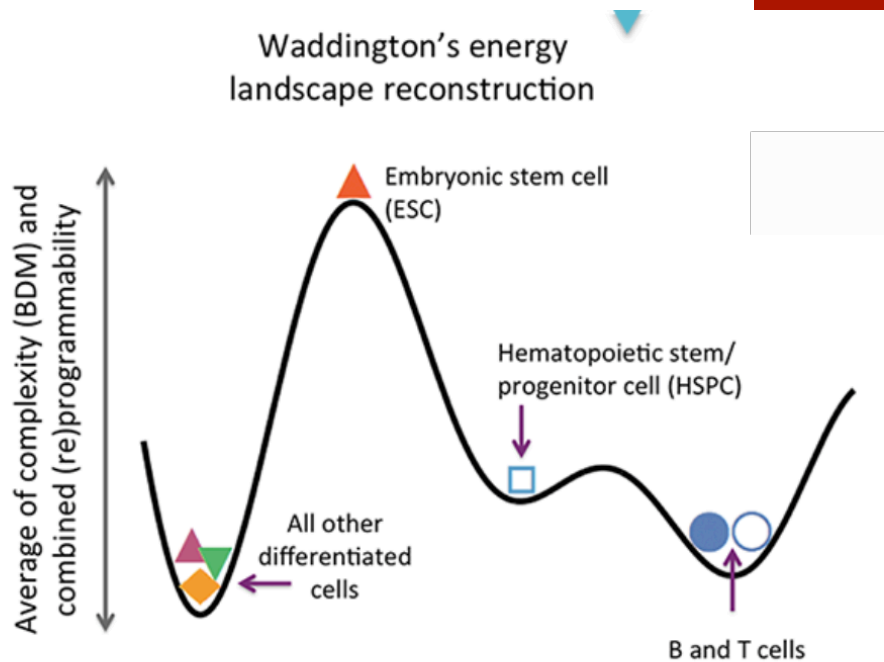


Info nodes of a T_h17 mouse cell in its differentiation process (gene expression, siRNA). A: Distribution of info genes. B: Non-uniform distribution of T_h17 marker genes. C: Information change in the 3 time steps. D: Overal network complexity.

An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Waddington's Landscape Reconstruction



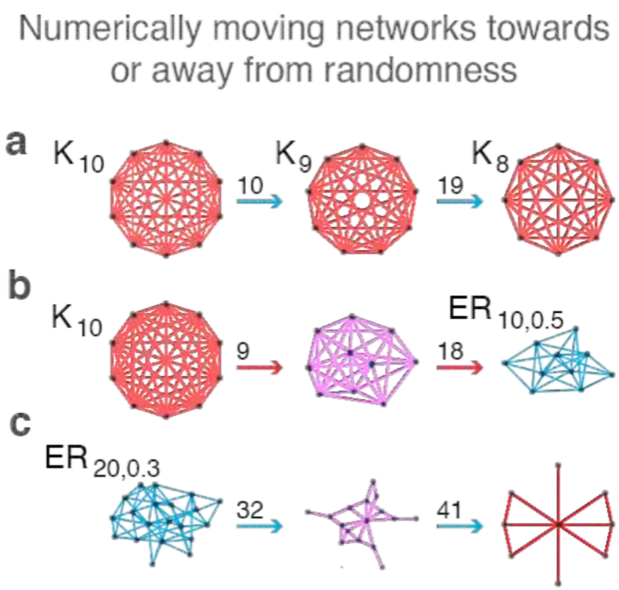


Data from: CellNet (Harvard)
Reconstruction of mammalian gene regulatory networks from 21 cell types and tissues: https://www.cell.com/cell/abstract/S0092-8674(14)00934-9

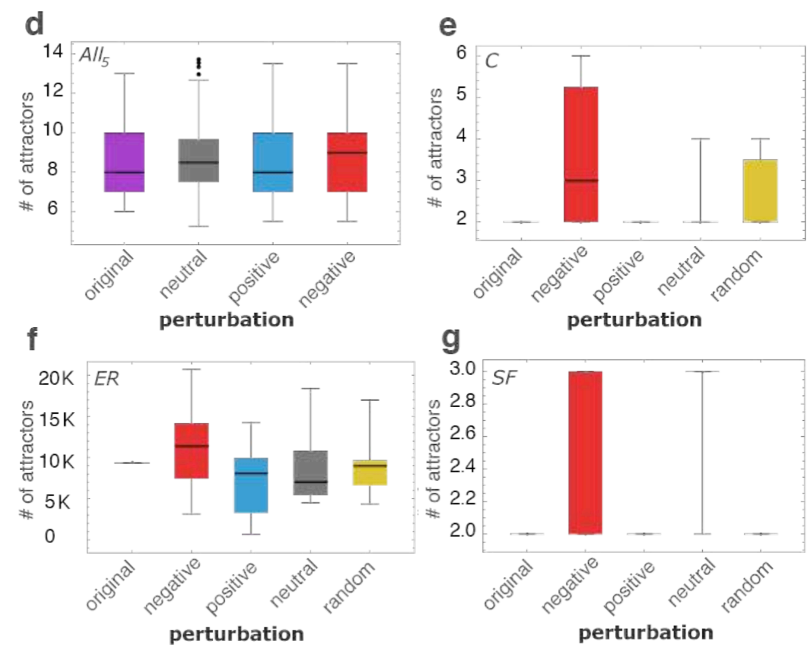
An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Algorithmic/Dynamic Landscape Relationship

Algorithmic space

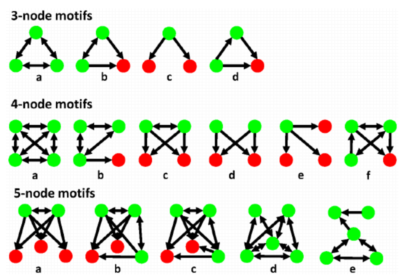


Dynamical phase space

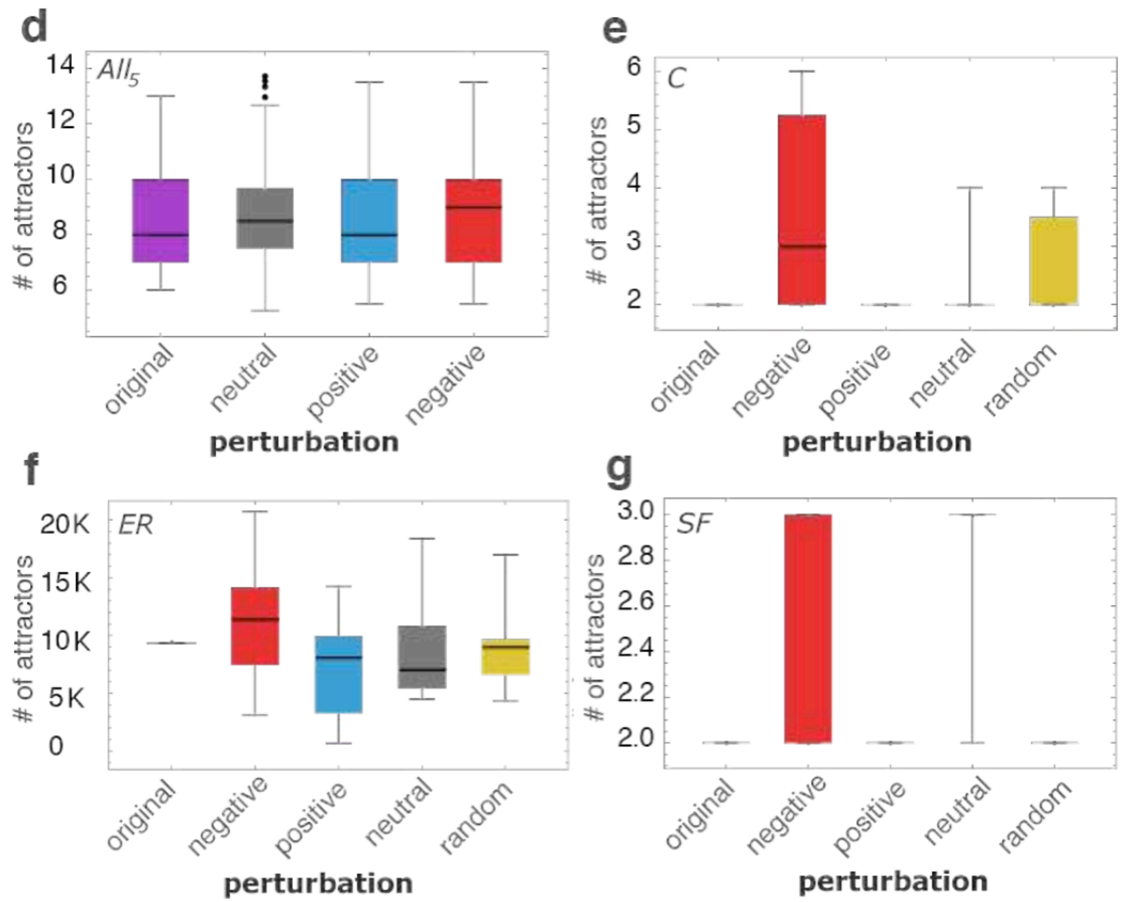


An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Test: all possible Boolean nets of size up to 5 & larger BNs

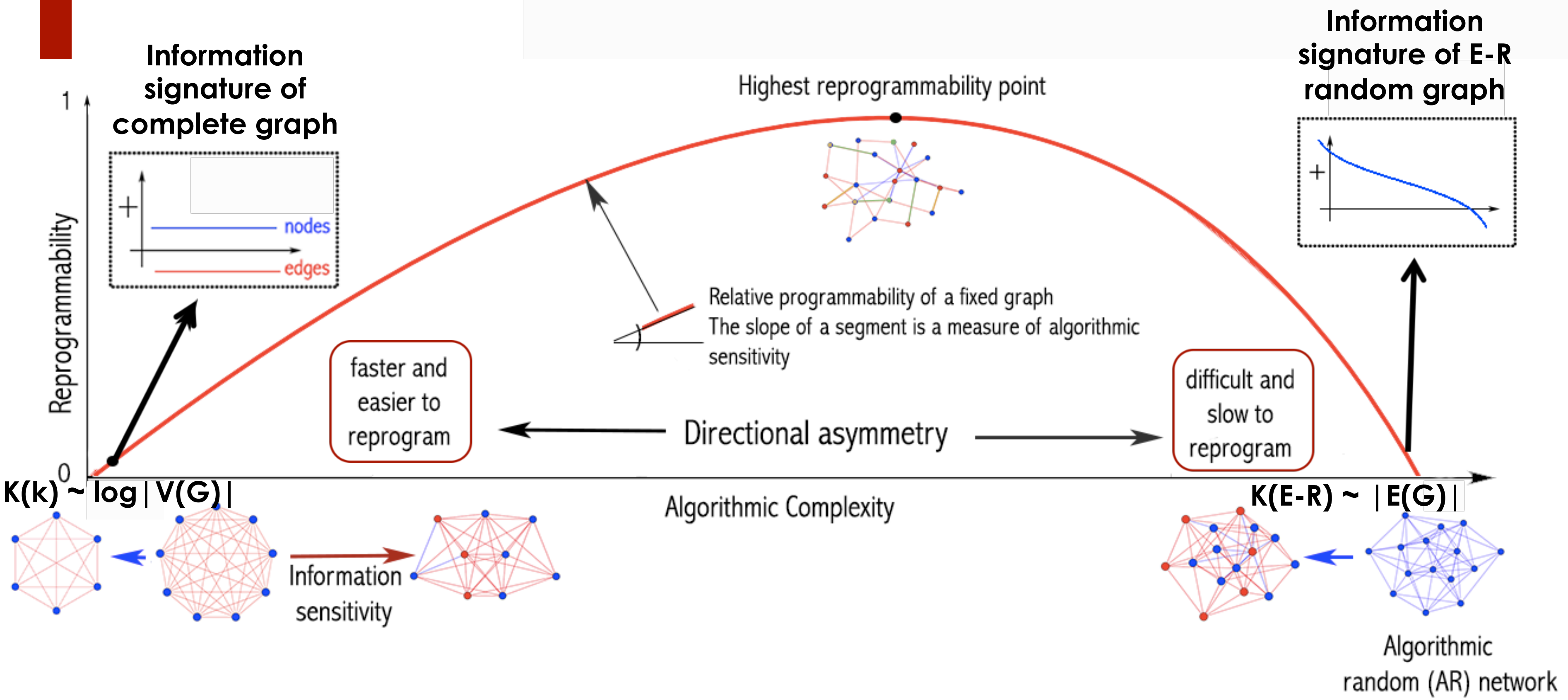


Perturbation by edge deletion



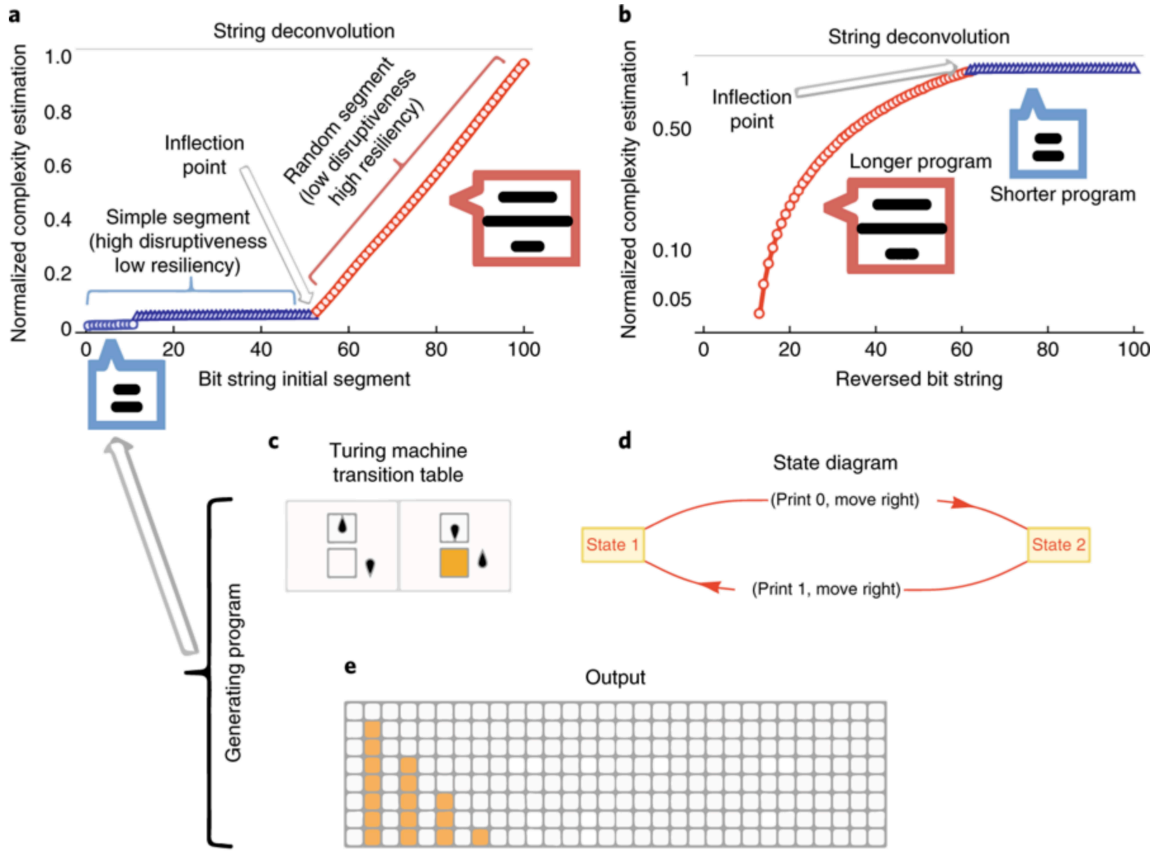
An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Thermodynamics of computer programs



An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems
H Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér
doi: https://doi.org/10.1101/185637

Algorithmic Deconvolution

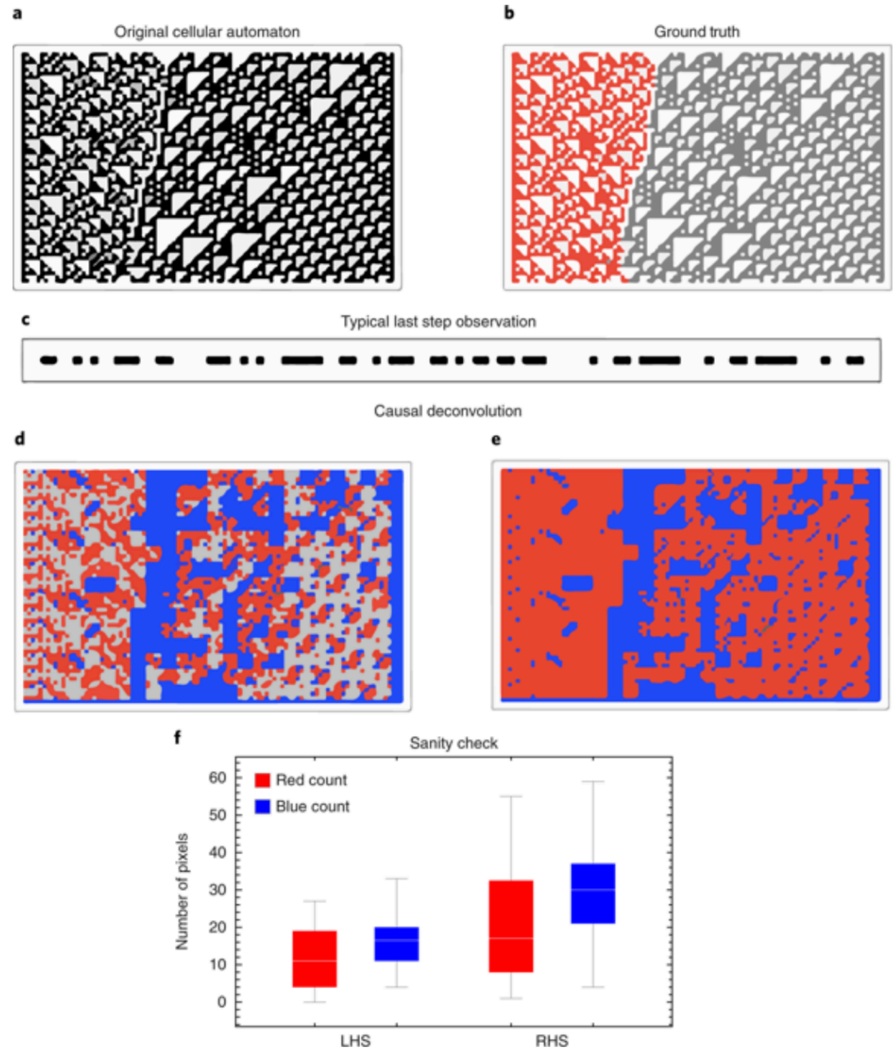


Sum of program-length minimization when breaking into smaller pieces

If a piece e is removed from a larger piece E and and $K(E\backslash e) < \log_2|e|$ then the likelihood that e is part of E is greater.

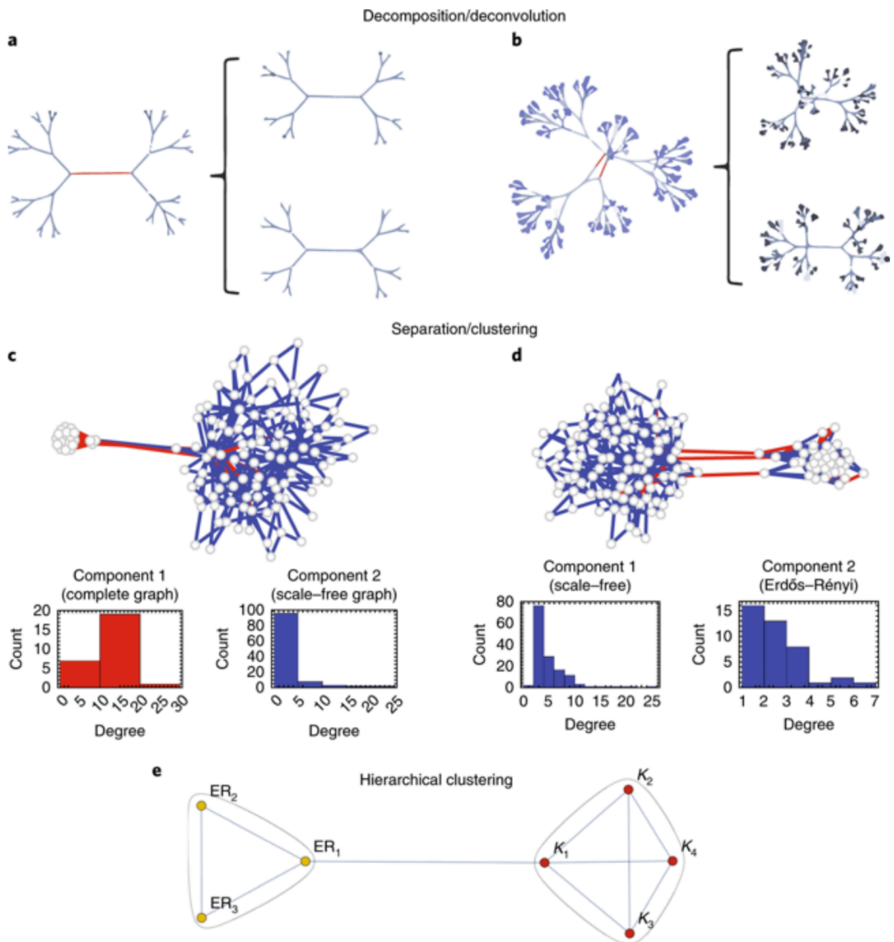
H. Zenil, N.A. Kiani, A. Zea, J. Tegnér, Causal Deconvolution by Algorithmic Generative Models
Nature Machine Intelligence, vol 1(1), pp 58-66, 2019.

CA separation



H. Zenil, N.A. Kiani, A. Zea, J. Tegnér, Causal Deconvolution by Algorithmic Generative Models
Nature Machine Intelligence, vol 1(1), pp 58-66, 2019.

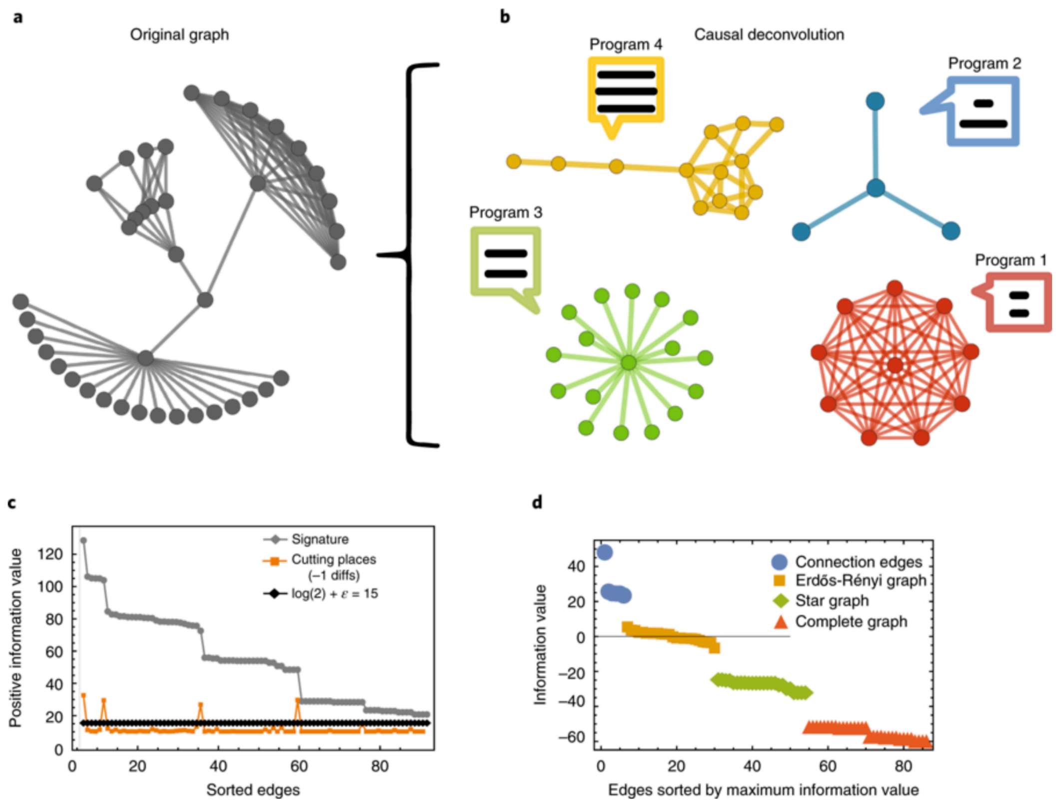
Network separation by generative mechanism



Breaking points are where the parts can be explained by the smallest sum of lengths of underlying computer programs

H. Zenil, N.A. Kiani, A. Zea, J. Tegnér, Causal Deconvolution by Algorithmic Generative Models **Nature Machine Intelligence**, vol 1(1), pp 58-66, 2019.

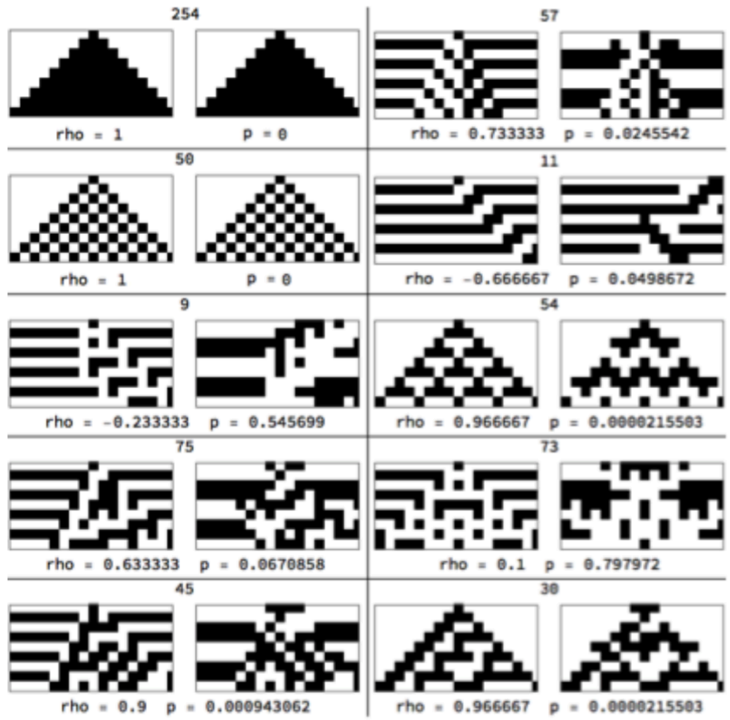
Network separation by generative mechanism



Termination criterion: When continue breaking actually increases the sum of the program-lengths

H. Zenil, N.A. Kiani, A. Zea, J. Tegnér, Causal Deconvolution by Algorithmic Generative Models
Nature Machine Intelligence, vol 1(1), pp 58-66, 2019.

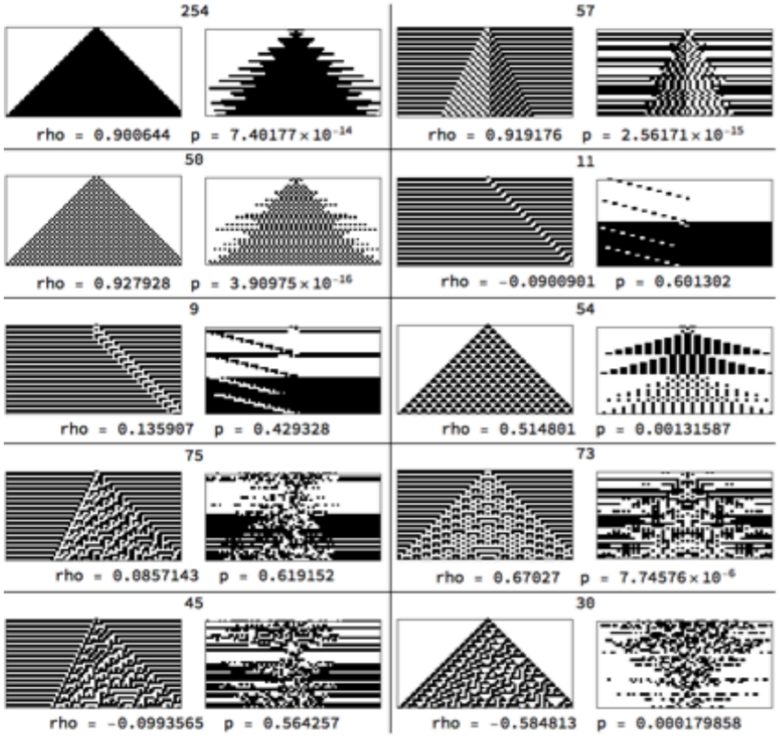
Reconstructing space-time diagrams by reconstruction of minimal K configuration



Lowest algorithmic complexity configuration of CA space-time diagrams.
Left columns: Original Right: Reconstructed.

H. Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér,
An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems, 2017

Reconstruction of time indexes (and thus, e.g. initial conditions & even the generating rule!)



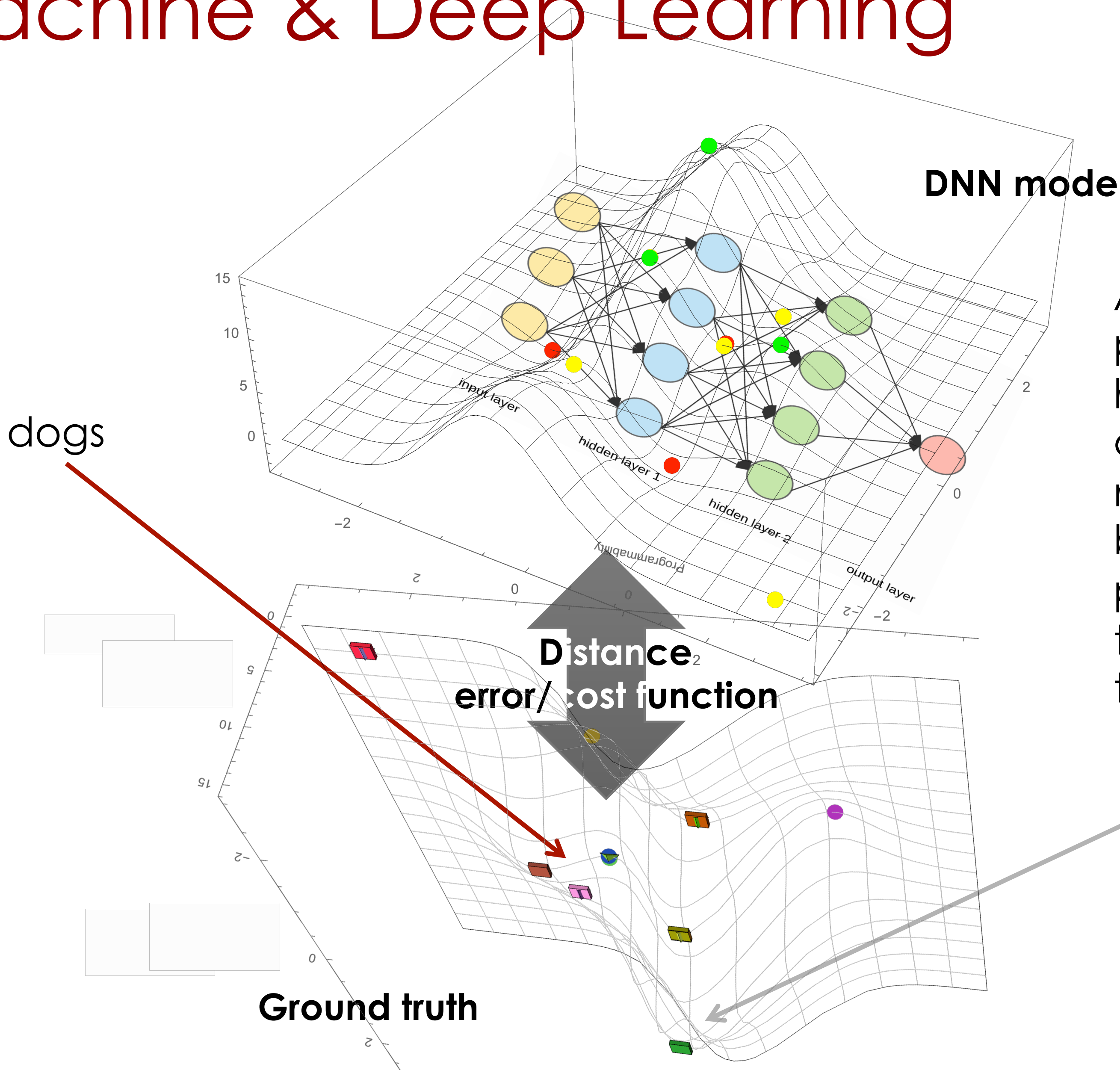
Intuition: You can recover the last step from the previous ones, so you can 'peel' back the system.

(we are applying this to continuous dynamical systems via K-Sinai entropy)

Adding time index to the reconstruction by perturbation analysis. Each row has a time index. Real vs guessed index Pearson correlation values reported.

H. Zenil, N.A. Kiani, F. Marabita, Y. Deng, S. Elias, A. Schmidt, G. Ball, J. Tegnér, An Algorithmic Information Calculus for Causal Discovery and Reprogramming Systems, 2017

Entropy at the core of Machine & Deep Learning

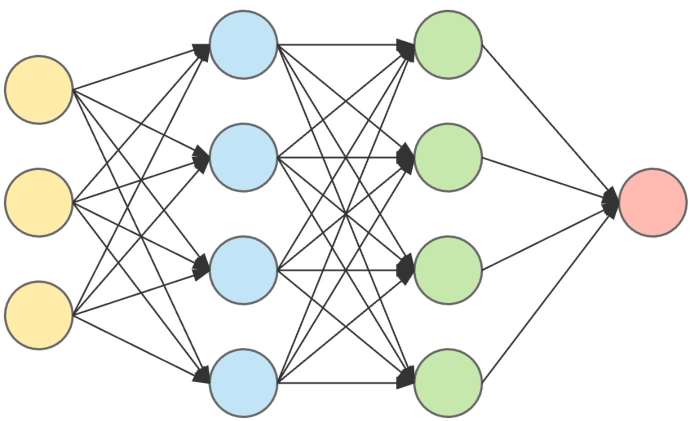


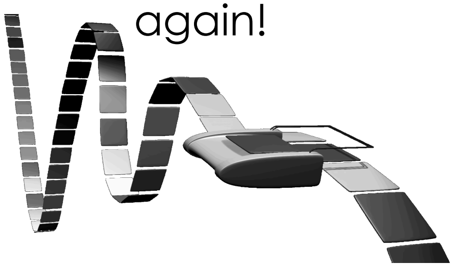
A DNN performance is how well the distance is minimised between every point in the training set vs the testing set

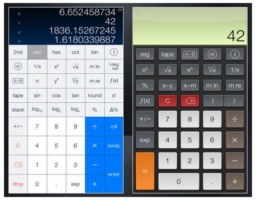
How do we fix this? Statistical + Symbolic

Paradoxically!
Back to square 1

We need to teach computers how to count again!







Statistical pattern matching:

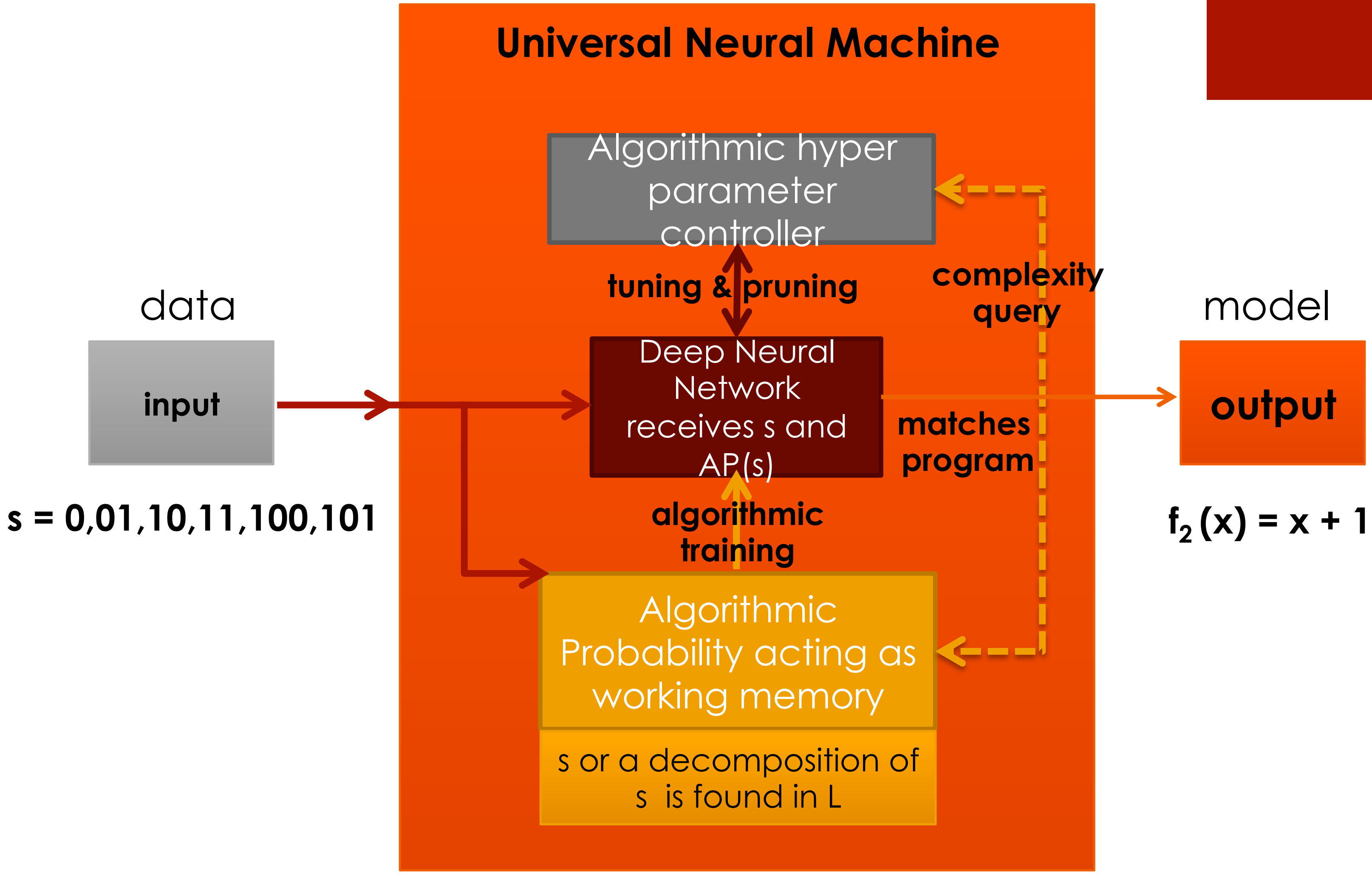
- Powerful statistical engine for pattern recognition
- Combinatorial and numerical data representation

Symbolic computation:

- Optimal inference engine
- Program/model synthesis
- Computational mechanics
- Algorithmic probability

Oxford Immune Algorithmics

Algorithmic Machine Learning



Three papers submitted to NeurIPS 2019