Theory Theory—
How We Come to Understand Our World

James P. Crutchfield
www.santafe.edu/∼chaos

3 May 2002

Abtract We are confronted, as never before, with an explosion in the size of databases: from data-mining the web to dynamical brain imaging, the various genome projects, and monitoring the millions of shipping containers that enter our ports. These datasets are so vast that there is no conceivable way humans can directly examine all of the data. What has been an intuitive and very human discovery process—hypothesizing structures and building predictive theories about the systems that produce them—must now be automated. But do we understand the notions of structure and pattern well enough that we can teach machines to discover them? How do we build theories that capture patterns in useful and predictive ways? I will announce some very recent results on automated pattern discovery and theory building for cellular automata. The lessons hint at what a future "artificial" science might look like.

Joint work with Carl McTague.

Data Explosion:

- Neurophysiology: multiple neuron recordings ($>$ 100 neurons @ 1kHz)
- Web Data-Mining: 10-100 GB/day, multi-terabyte databases
- Astrophysical data: Hubble, EUV, ...
- Neuroimaging: MEG 100-200 Squids @ 1 kHz
- Geophysics: Earthquake monitoring with 1000s sensors, of different kinds
- BioInformatics: Genome projects, microarray sequencing, ...
- Searchable Video Databases
- Very large-scale simulations: weather, hydrodynamics, reaction kinetics, ...
- ...

Need machines to help.

Current approach is Pattern Recognition:

- Match data against existing palette of templates

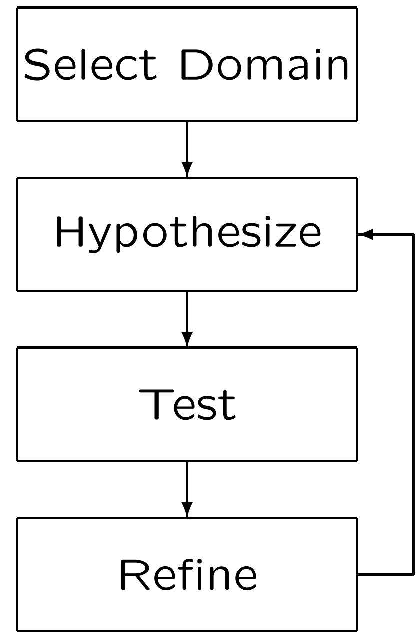
- Fitting polynomials, assuming IID and distributions are Gaussian, taking Fourier/Laplace Transforms, ...

Begs the Question:

Where did that palette come from in the first place?

How is this done now?

Baconian Scientific Algorithm:



Hypothesis generation: Guessing within a discipline's domain. Must we always "guess" or intuit?

Problem: Hypothesis wrong? Little hint of where to go next.

The Inverse Problem: How to go from Data to Theory?

History of the Inverse Problem

Recently in nonlinear physics,

Attractor reconstruction:

- Packard et al (1980): "Geometry from a Time Series"
- Takens (1981), ...

"Theory From Time Series":

- JPC/McNamara (1987): Equations of Motion from a Data Series
- Farmer/Sidorovich (1987): Nonlinear prediction
- Casdagli (1989), ...

Problem with this, too, alas.

The Underlying Problem: Intrinsic Representation

Reformulating the Question:
Does data contain information about an appropriate representation?

Solution:
Tackle head on: Define structure, pattern, regularity, ...

Computational Mechanics:
Accounts for computational structure.
Extends Statistical Mechanics: more than “statistics”.

Analogy:

Physics "accounts" for energy flow and transduction.

Computational Mechanics accounts for

1. Amount of historical information stored in process;
2. The architecture supporting that storage; and
3. How stored information is transformed into future states.

More directly:

1. Physics has measures of disorder: temperature, thermodynamic entropy.
2. Comp'l mechanics measures degrees of pattern: structural complexity.

Causal Architecture: A Review of Computational Mechanics

Processes: $\overleftrightarrow{S} = \ldots S_{-1} S_0 S_1 \ldots = \overleftarrow{S}\overrightarrow{S}$.

Causal States $\mathcal{S}$: Sets of $\overleftarrow{S}$ that are equally predictive about $\overrightarrow{S}$

$$\overleftarrow{s} \sim \overleftarrow{s}' \text{such that } \mathrm{P}(\overrightarrow{S} \mid \overleftarrow{S} = \overleftarrow{s}) = \mathrm{P}(\overrightarrow{S} \mid \overleftarrow{S} = \overleftarrow{s}') \qquad (1)$$

ϵ-Machine: $M = \{\mathcal{S}, \mathcal{T}\}$

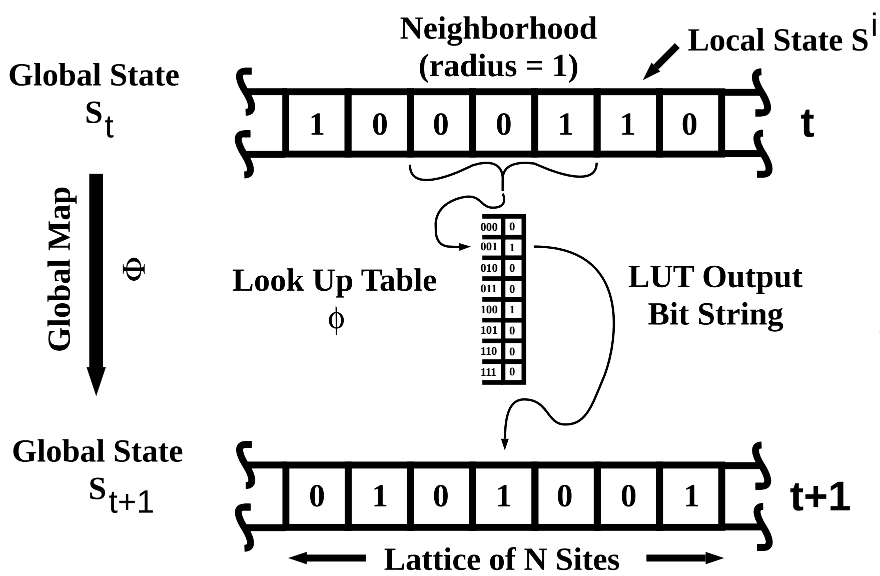
- Theorem: M is the unique, optimal predictor of minimal size.
- Theorem: M is a minimal sufficient statistic for a process.
- The *intrinsic representation* of a process.

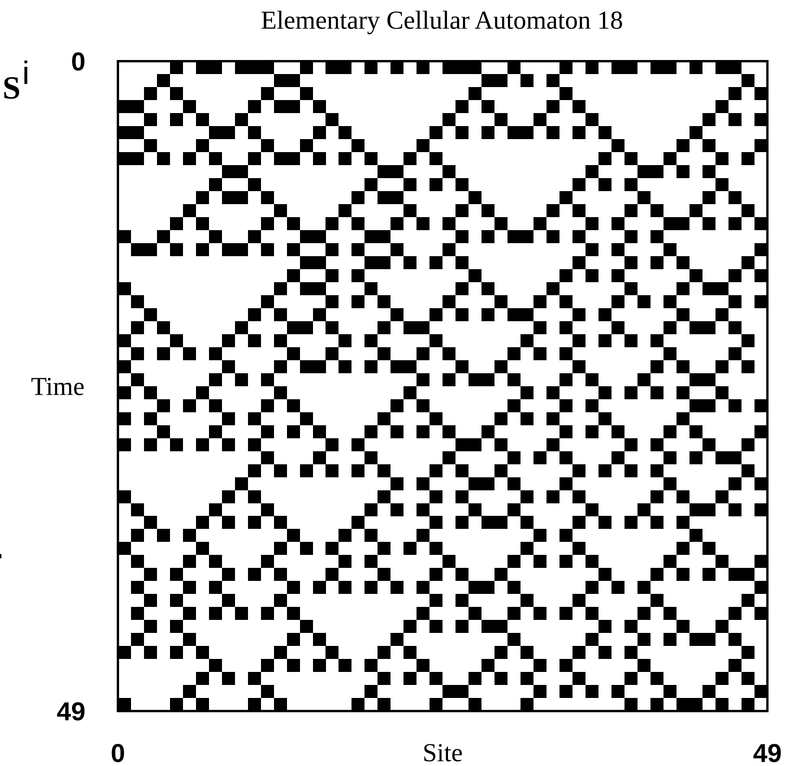
Stored information is Statistical Complexity: $C_\mu = H[\mathrm{P}(\mathcal{S})]$.

JPC & K Young, Physical Review Letters **63** (1989) 105–108.

CR Shalizi & JPC, J. Statistical Physics **104** (2001) 817–879.

Cellular Automata: Artificial Physics's

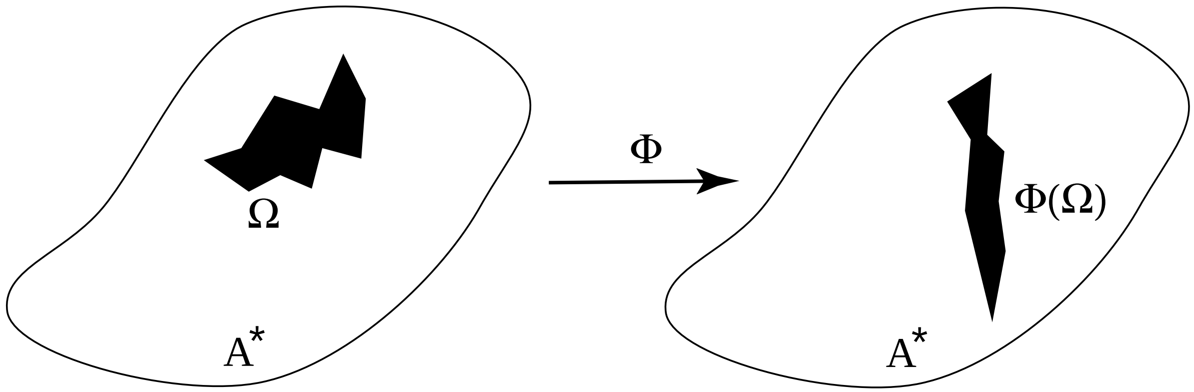




ECA 18 Rule Table:

η_t	000	001	010	011	100	101	110	111
$s_{t+1} = \phi(\eta_t)$	0	1	0	0	1	0	0	0

Cellular Automata Computational Mechanics



Space of configurations $\mathcal{A}^*$: Represented by finite-state automata.

CA LUT: A finite-state transducer T_Φ that reads in sets Ω_t and outputs Ω_{t+1}.

Language evolution:
Implemented by Finite-Machine Evolution (FME) Operator Φ:
(i) $\Phi(\Omega)$, (ii) Drop Transducer Inputs, (iii) Strip Transient States, (iv) Convert NFA $\rightarrow$ DFA, and (v) Minimize.

Computational Complexity, starting with n-state language:

- $\Phi(\Omega)$ and Drop Inputs: $O(n)$.
- Identify Transient States: $O(n^2)$.
- NFA $\rightarrow$ DFA: $O(2^n)$.
- Minimize: $O(n \log_2 n)$.

Wolfram (1984): Iterate all configurations $\Omega_0 = \mathcal{A}^*$:

$$\Omega_t = \Phi^t \Omega_0 \ . \tag{2}$$

Regular Language Complexity: $|\Omega_t|$

t	ECA 18	ECA 22
1	5	15
2	47	280
3	143	4506
4	$\geq 20,000$	$\geq 20,000$

A Structural View

Domain: Spacetime shift-invariant pattern

1. $\Lambda = \Phi^p(\Lambda)$, for some $p > 0$.

2. $\Lambda = \sigma^m(\Lambda)$, for some $m > 0$.

Particle:

1. $\Lambda^i \alpha \Lambda^j$, $\alpha \in \mathcal{A}^*$.

2. $\alpha = \Phi^p(\alpha)$, for some $p > 0$.

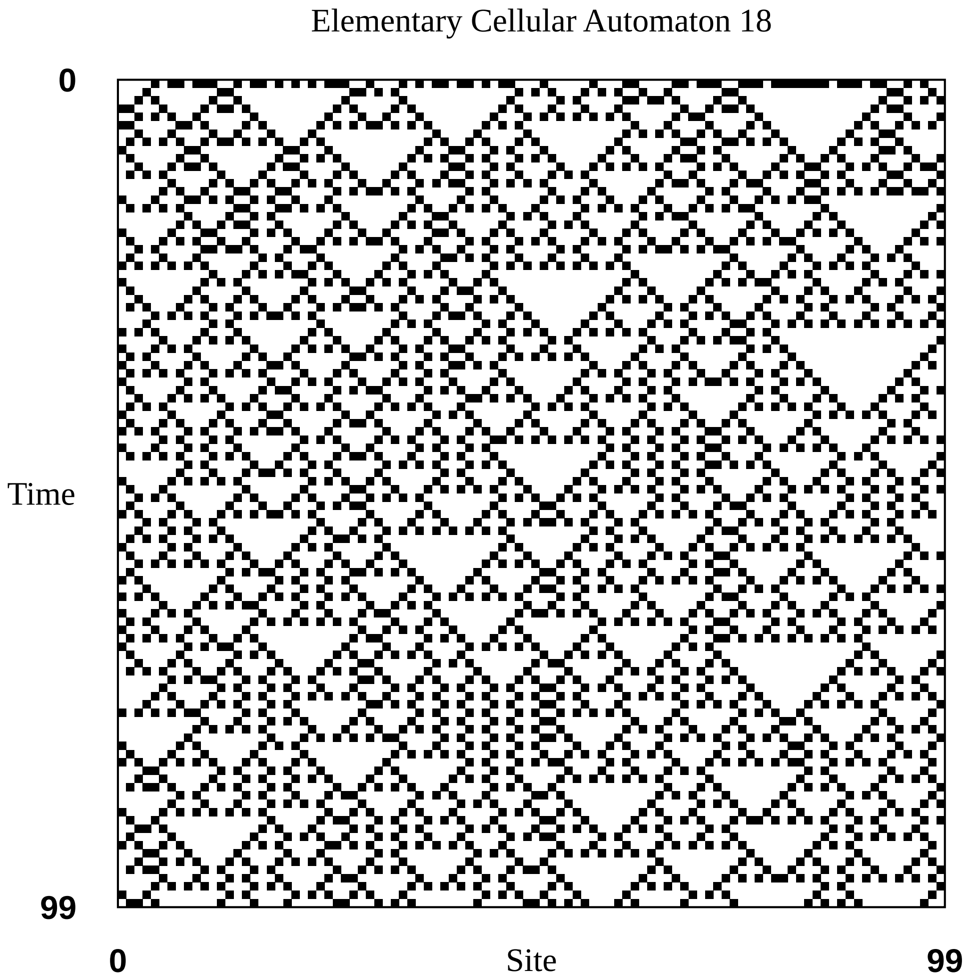
Particle Interactions:

1. $\alpha + \beta \rightarrow \gamma$.

2. $\alpha + \nu \rightarrow \emptyset$.

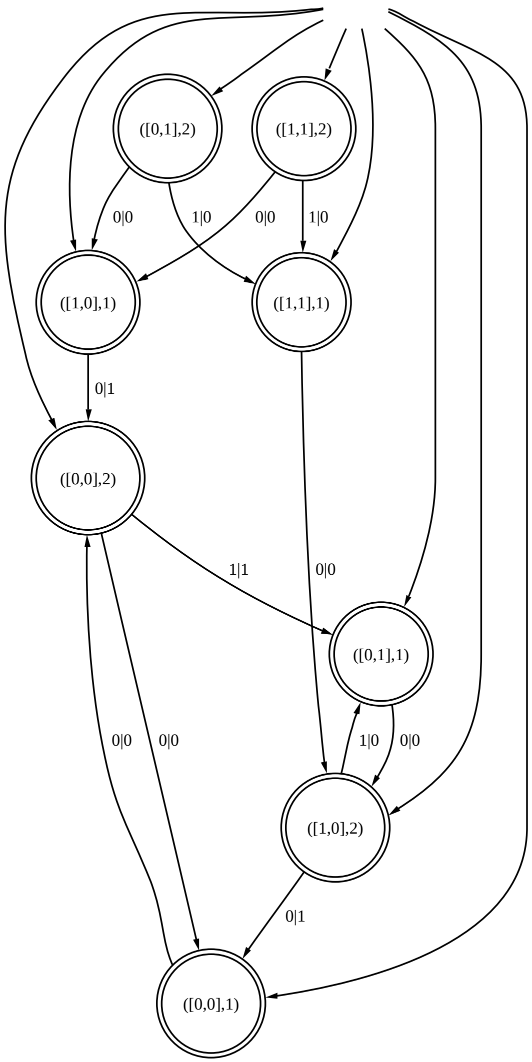
3. ....

Example: ECA 18



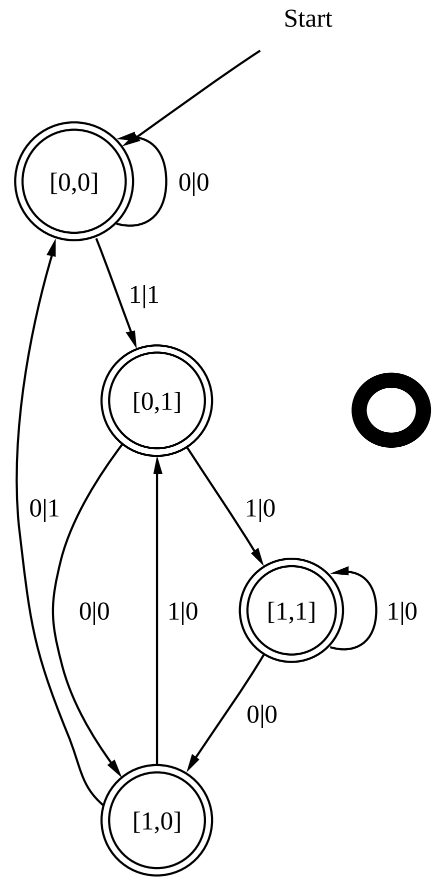
Regularity: Regions of $(0\mathcal{A})^*$ ... A domain?

Iteration of ECA 18's Domain



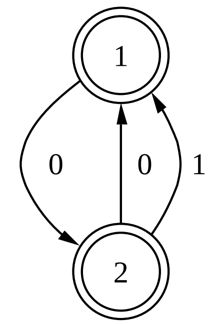
Composition

=

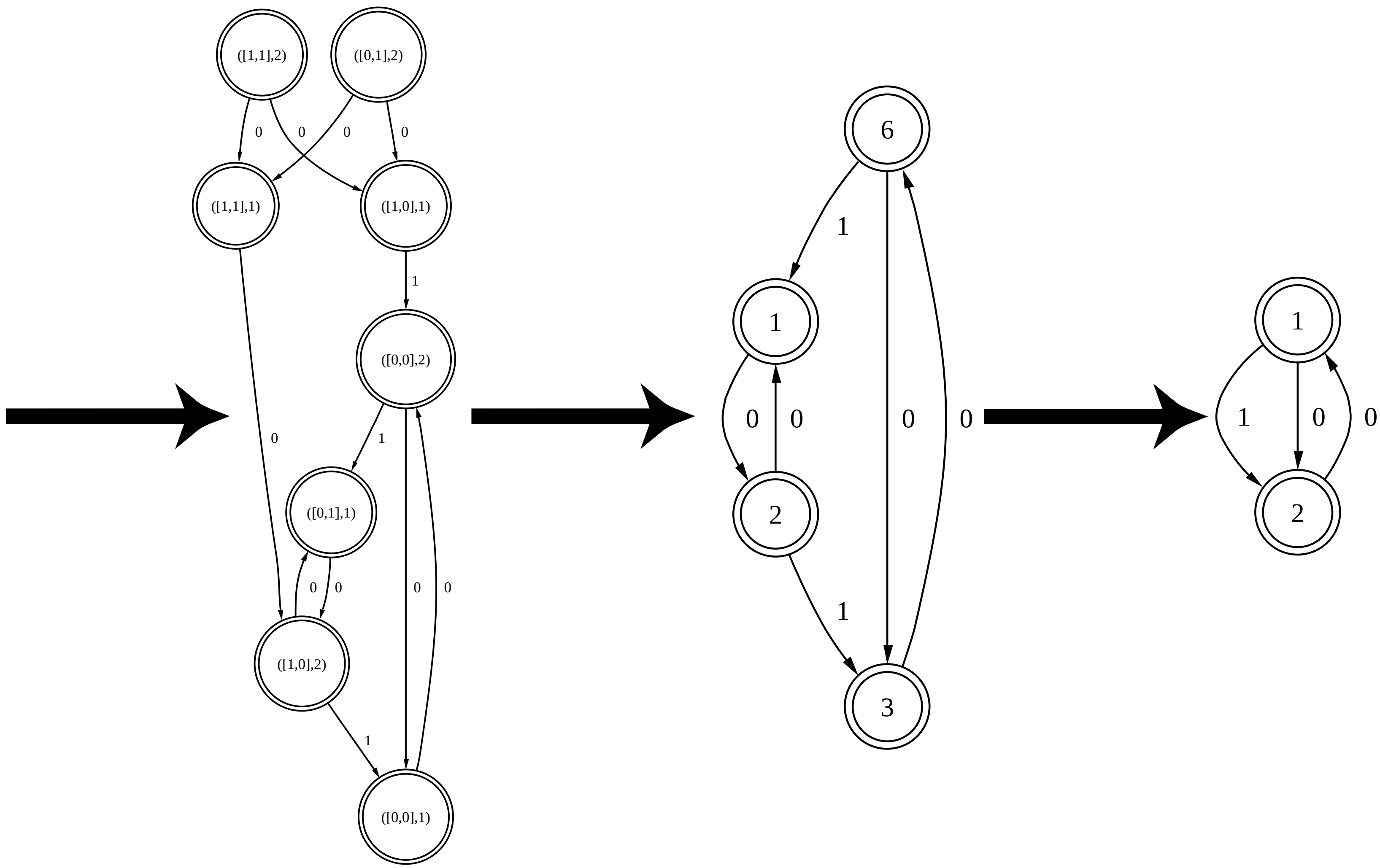


ECA 18 Transducer

o



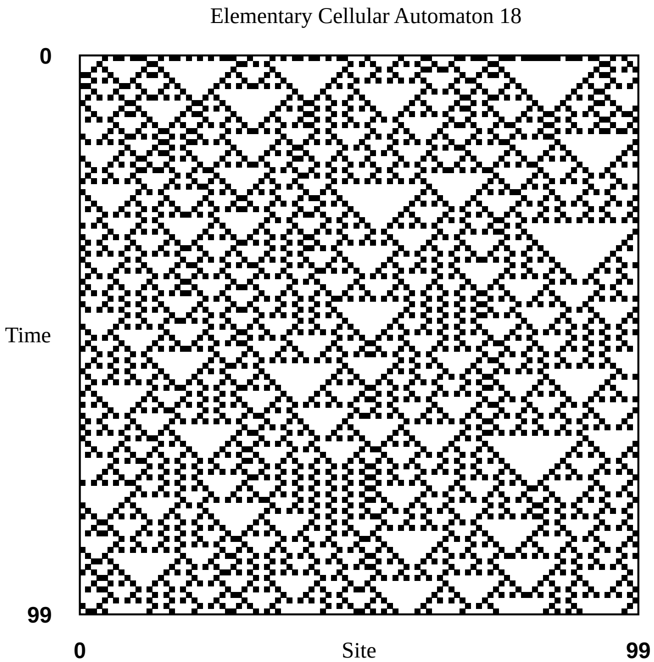
Hypothesized Domain

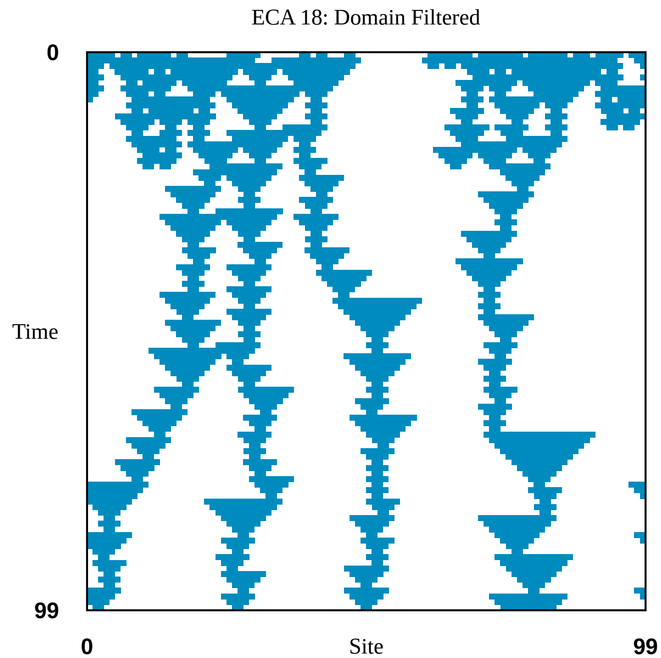


Drop Inputs   Strip Transients   Minimize   QED

Recognizing Structure: Domain Filtering

Factor out that data explained by structures one knows.

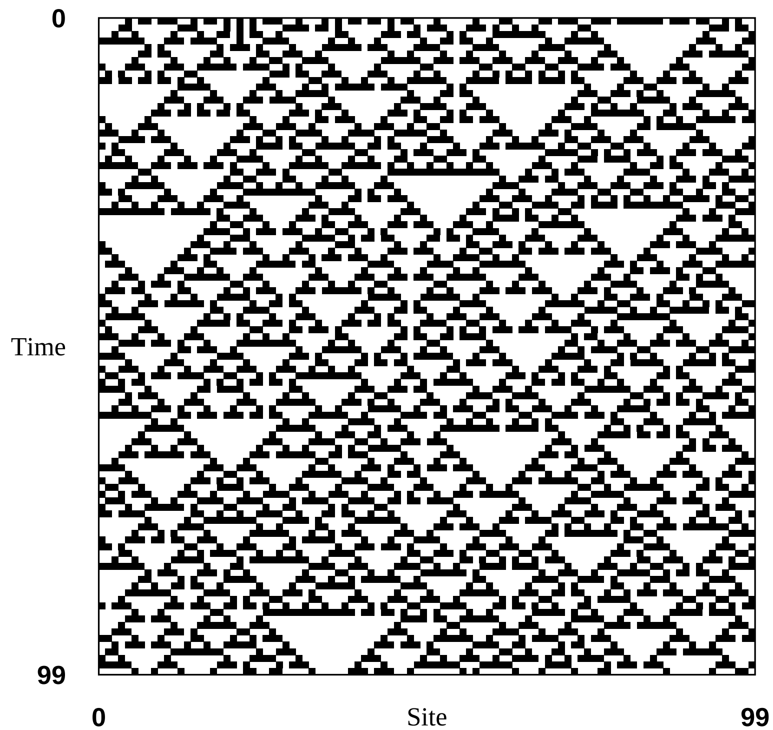




ECA 18's Particle Catalog	
Domain	Process Language
Λ^0	$(0\Sigma)^*$
Particle	Wedges
α	$\Lambda^0 0^{2n+1} 0 \Lambda^0$
Interactions	
$\alpha + \alpha \rightarrow \emptyset$	

The Enigma: ECA 22

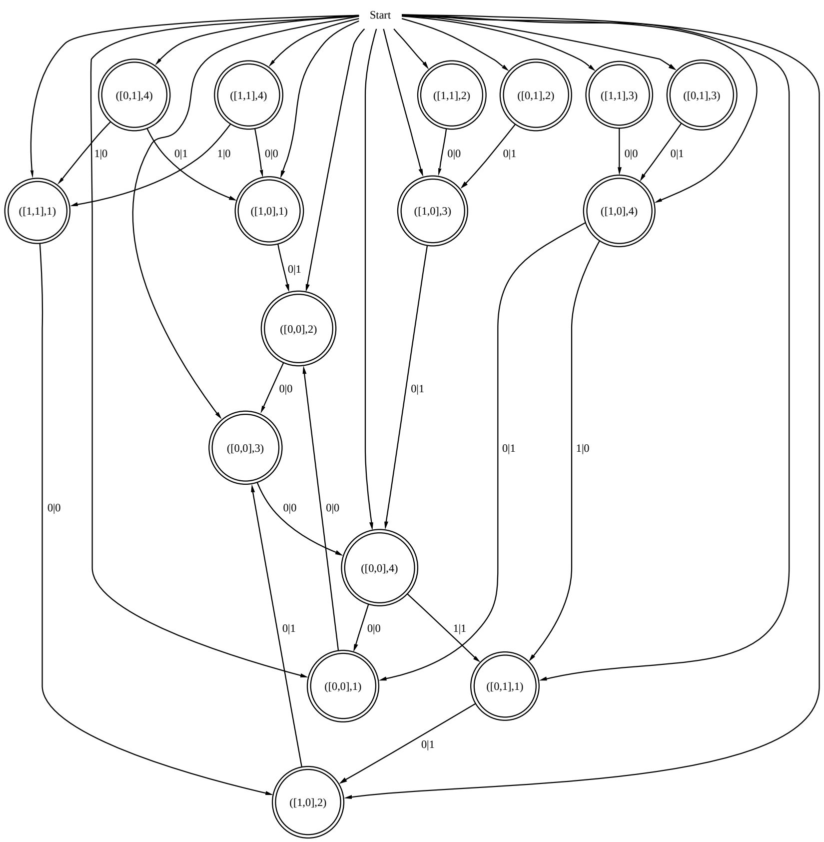
ECA 22 Lookup Table								
η_t	000	001	010	011	100	101	110	111
$s_{t+1} = \phi(\eta_t)$	0	1	1	0	1	0	0	0



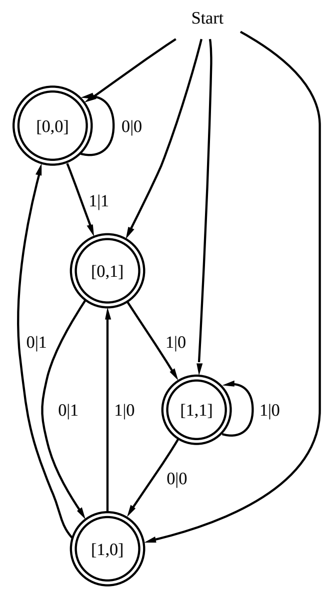
History: highly disordered (high entropy) but ...

- Moore: "... nonlocal structures ..."
- Grassberger (1986): "... long-range, complex correlations ..."; Power law decays.

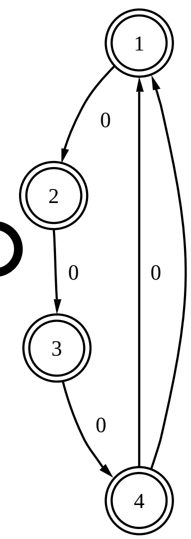
Iterate of ECA 22's Candidate Domain



=



∘

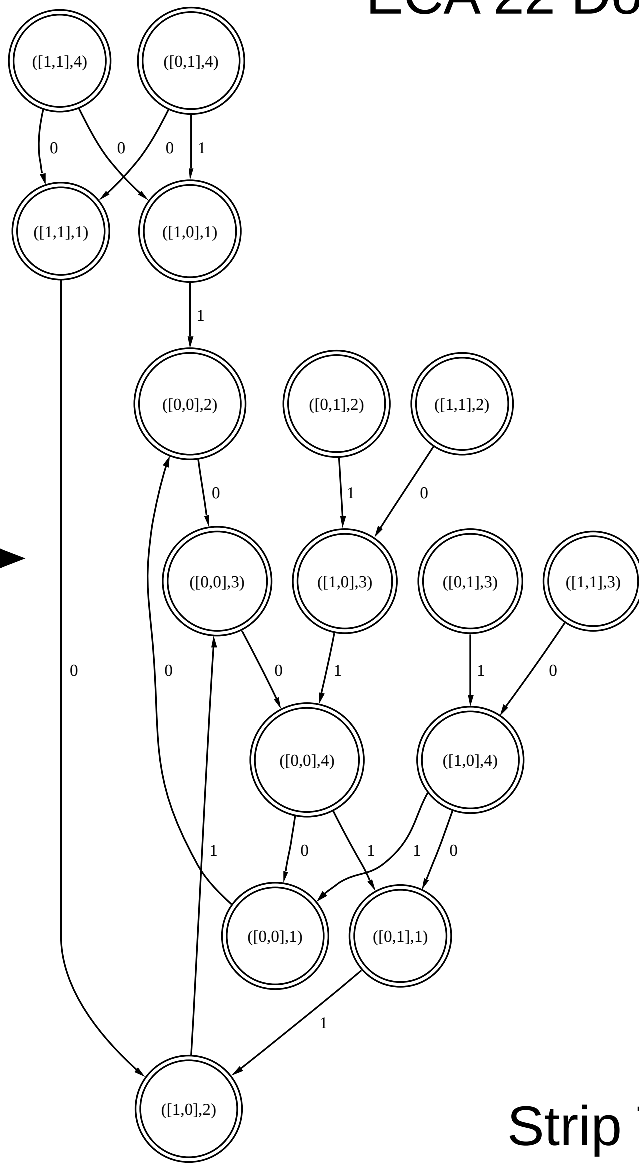


Composition

Transducer

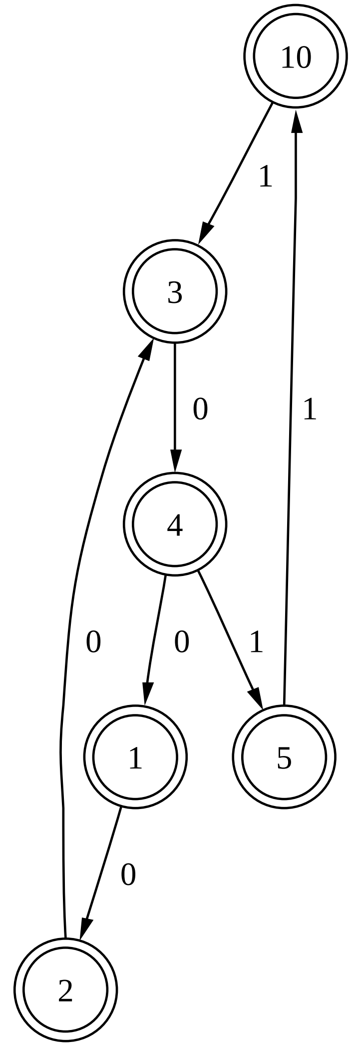
Candidate Domain

ECA 22 Domain Proof?



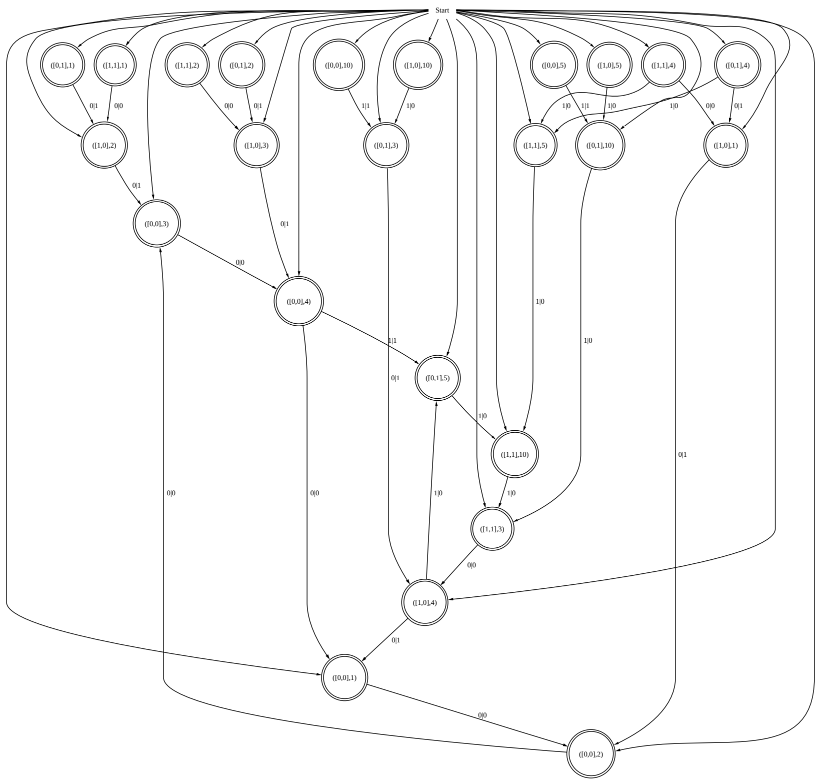
Drop Inputs

Strip Transients
& Minimize



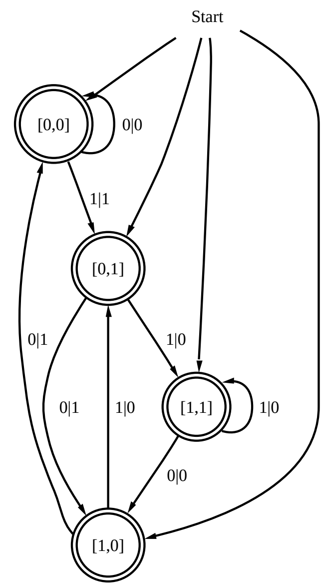
But ≠ Candidate!

Second Iterate of ECA 22's Candidate Domain



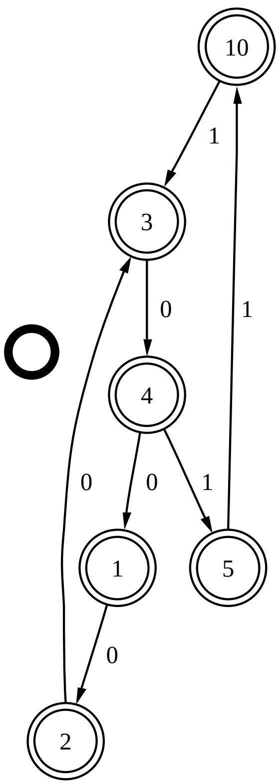
Composition

=



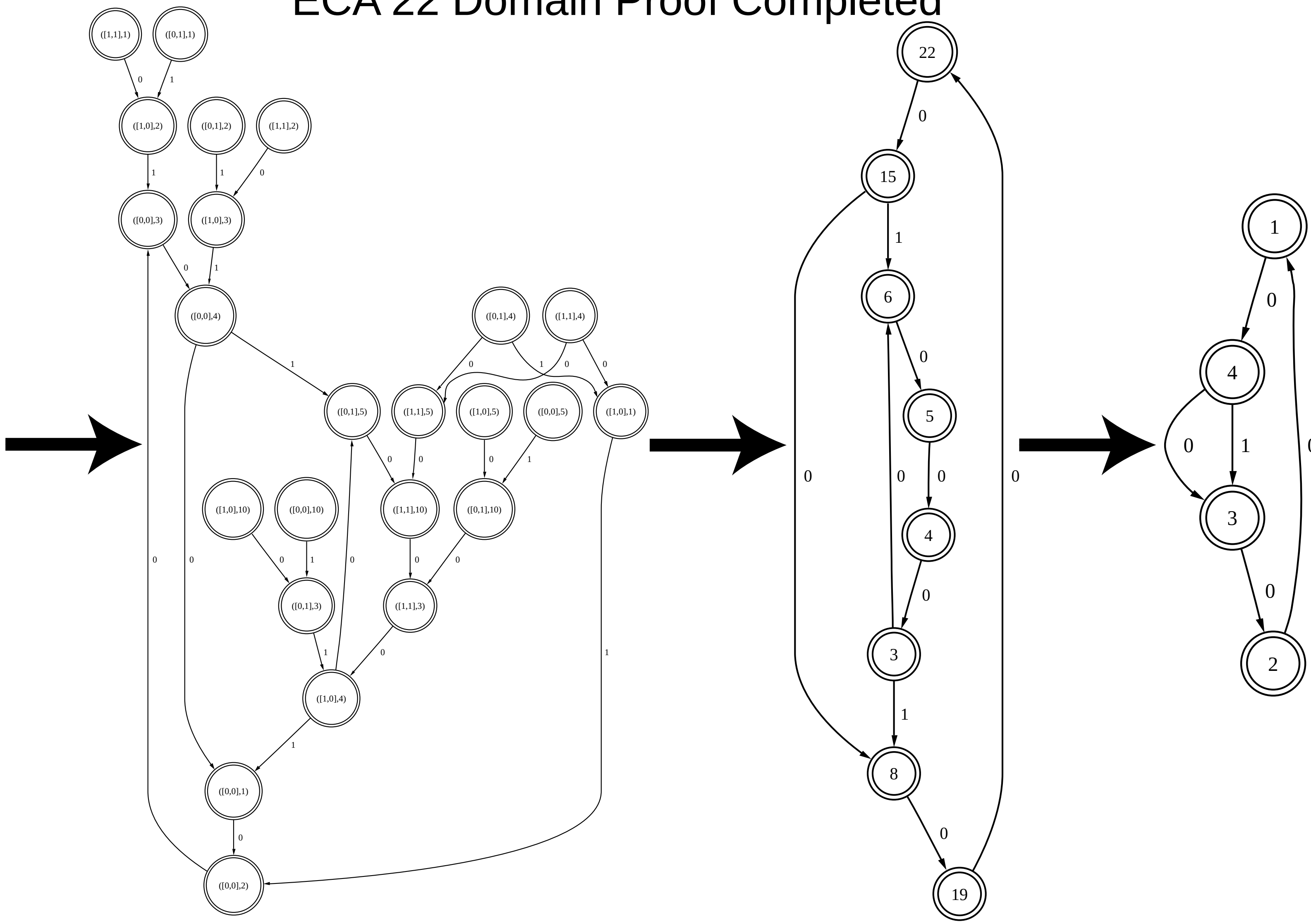
Transducer

o



Candidate Iterate

ECA 22 Domain Proof Completed



Drop Inputs

Strip Transients

Minimize QED!

Theorem: $\{\Lambda_0^0, \Lambda_1^0\}$ is a spacetime domain for ECA 22.

Proof:

(i) $\Lambda_1^0 = \Phi(\Lambda_0^0)$

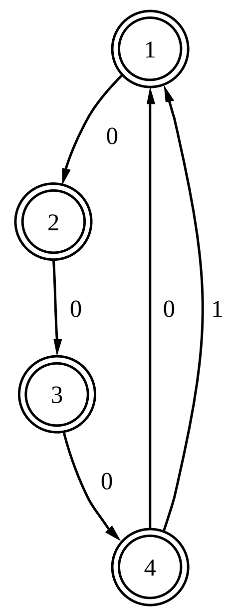
(ii) $\Lambda_0^0 = \Phi(\Lambda_1^0)$

in other words

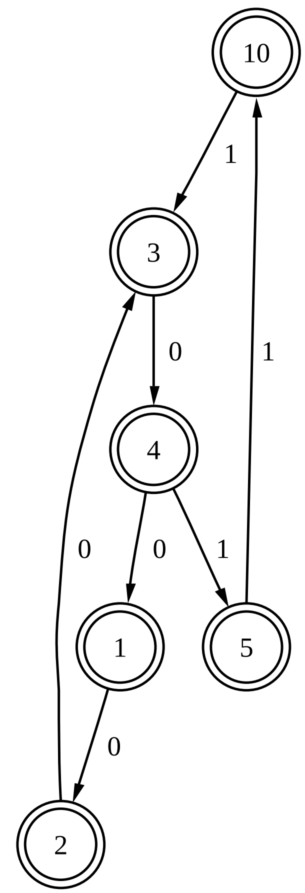
(iii) $\Lambda_0^0 = \Phi^2(\Lambda_0^0)$

(iv) $\Lambda_1^0 = \Phi^2(\Lambda_1^0)$

Λ_0^0

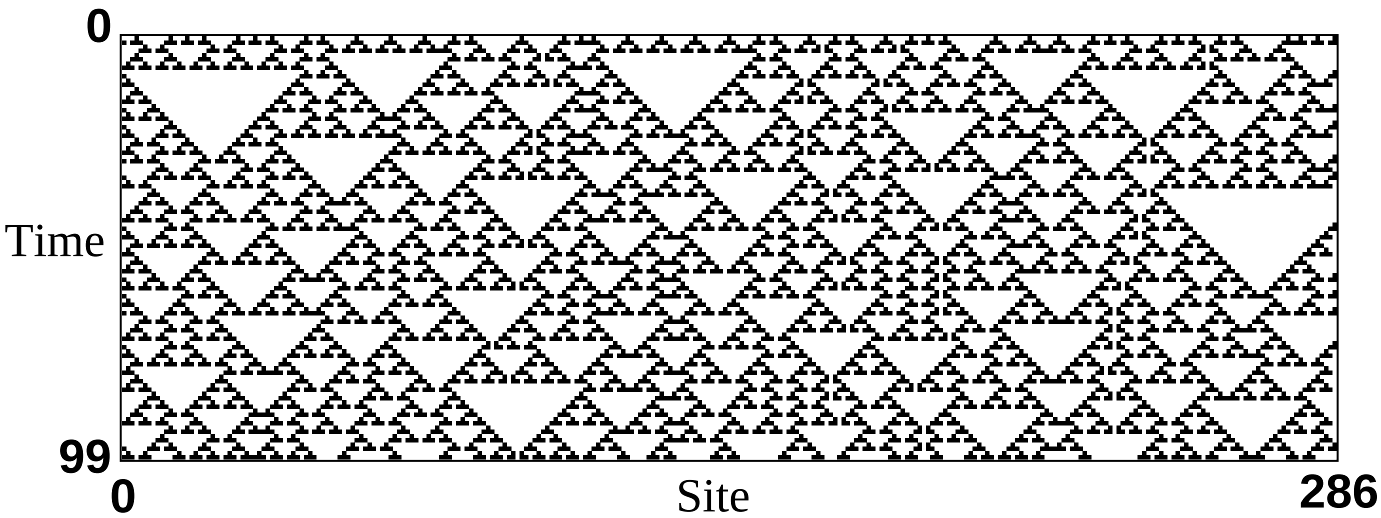


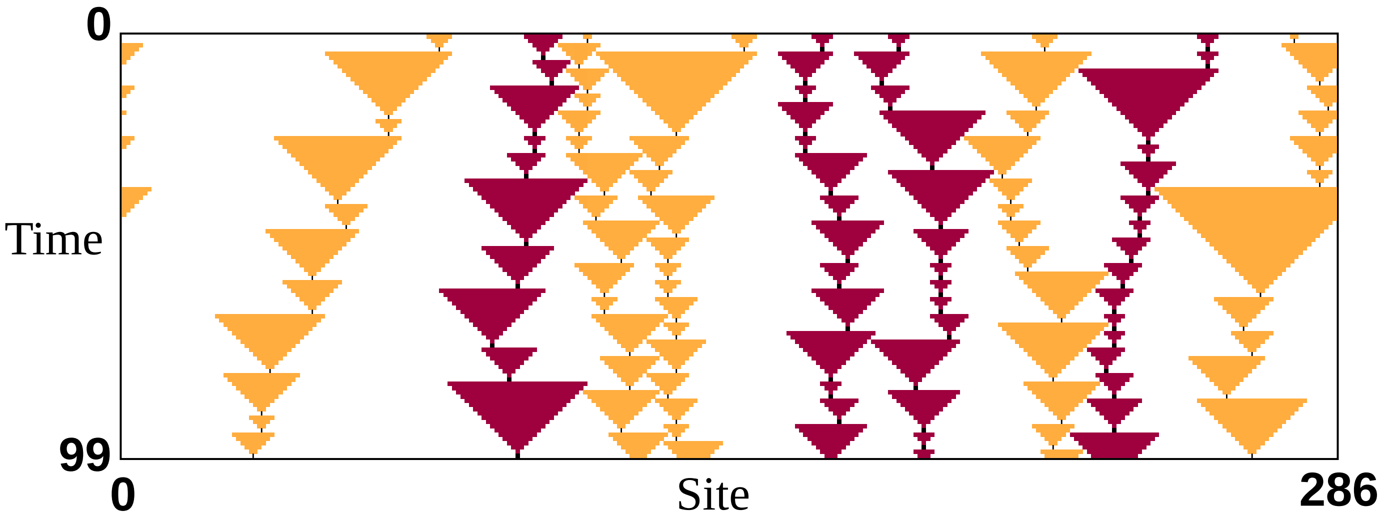
Λ_1^0



This is the first structure discovered for ECA 22.
Captures much of ECA 22's behavior.

The Enigma? ECA 22





ECA 22's Particle Catalog	
Domains	Process Language
Λ^0	$\Lambda^0_0 = (000\Sigma)^*$ $\Lambda^0_1 = (1110 + 0000)^*$
Λ^1	0^*
Λ^2	$(01)^*$
Λ^3	$(0011)^*$
Particles	Wedges
α	$\Lambda^0_{0A} 001_A \Lambda^0_0$ $\Lambda^0_{1A'} 1111_{B'} \Lambda^0_1$
β	$\Lambda^0_{0A} 01_A \Lambda^0_0$ $\Lambda^0_{1A'} 110_{B'} \Lambda^0_1$ $\Lambda^0_{1A'} 001_{B'} \Lambda^0_1$
Interactions	
$\alpha + \alpha \rightarrow \beta$ $\beta + \beta \rightarrow \emptyset$ $\alpha + \beta \rightarrow \Lambda_{\alpha-\beta \text{gas}}$ $\beta + \alpha \rightarrow \Lambda_{\alpha-\beta \text{gas}}$	

ECA 22: Pattern Discovery

Theorem: $\{\Lambda_0^0, \Lambda_1^0\}$ is the only (positive entropy) domain within the complexity horizon (up to and including 6-state candidates).

Proof: Automated proof methods using the Haskell, a functional programming language.

How was this done?

Candidates for Domains: Process Languages

Finite-state process language: represented by (recurrent portion of) ϵ-machine.

AKA finite-state machine with one strongly connected component; all states are start and final states.

n-DCL Library: The domain candidates with n-states.

How many are there?

States	n-DCL Library
n	Size
1	3
2	7
3	78
4	1,388
5	35,186
6	1,132,613

Automated Search

1. For all $\Lambda \in n$-DCL, exclude by counterexample:
   (a) Generate long $\mathbf{s} \in \Lambda$.
   (b) Iterate this single configuration $\Phi^t(\mathbf{s})$.
   (c) Test: If $\Phi^t(\mathbf{s}) \notin \Lambda$, then Λ is expanding.

2. For all nonexpansive n-DCL, exclude by entropy rate:

$$h_\mu(\Phi^t(\Lambda)) < h_\mu(\Lambda) \Rightarrow \text{Candidate contracts.} \quad (3)$$

3. For all nonexpanding, noncontracting n-DCL, attempt to directly prove invariance theorem: $\Lambda = \Phi^p \Lambda$?

Λ that pass tests are domains.

ECA 22: Table of a Million Theorems

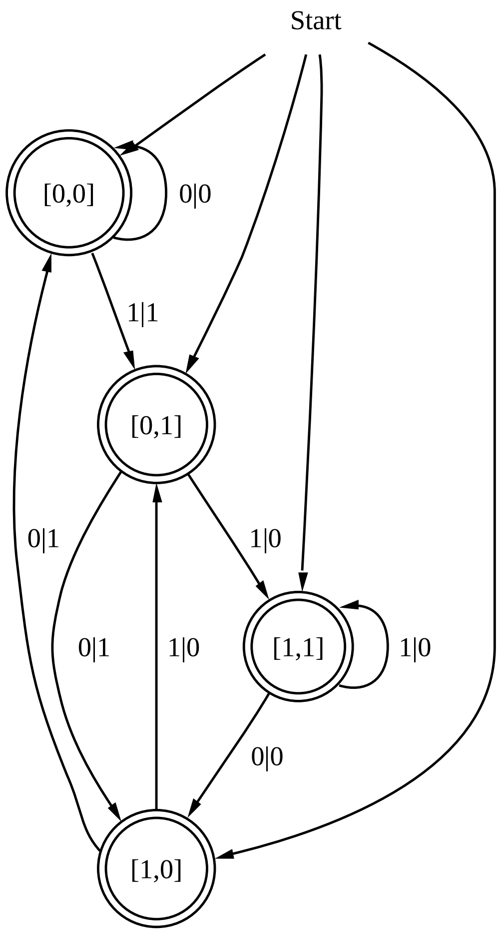
States	n-DCL	ECA 22 Nonexpanding n-DCL		
n	Size	Number	Contracting	Domains
1	3	2	Σ^*	$\Lambda^1 = 0^*$
2	7	2	$(0\Sigma)^*$	$\Lambda^2 = (01)^*$
3	78	0		
4	1,388	2		Λ^0_0, $\Lambda^3 = (0011)^*$
5	35,186	5	$((11)(11+01)^*(10)+00)^*$ $((101)(00+1)+(1(00+1)+0))^*$ $([(101)^+(1(00+1)+00)]+[1(00+1)+0])^*$ $([1^+(01)(00+1)]+[1^+(00)+0])^*$ $([(1^+01)^+(1^+00+00)]+[1^+00+0])^*$	
6	1,132,613	268	267	Λ^0_1

QED

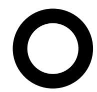
Several months ago: Estimated compute time $\sim$ 5 Beowulf years!

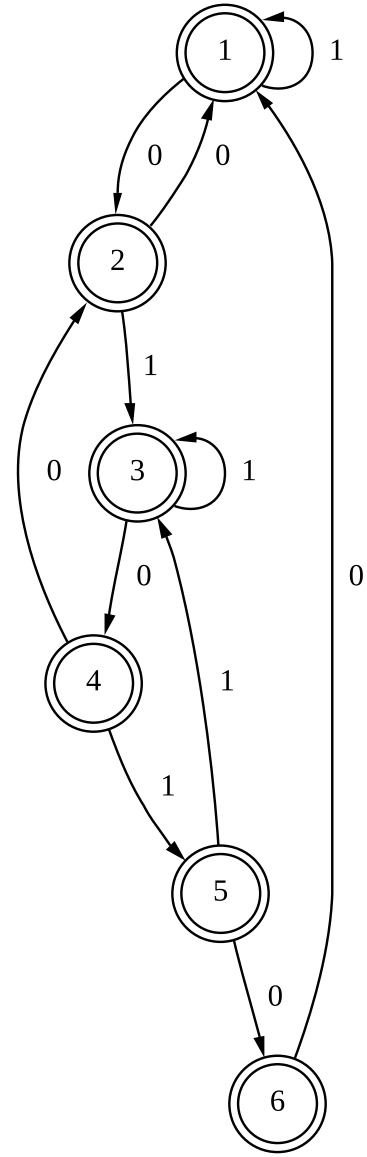
When finally proved, proof took $\sim$ 1 week: speed = 7000 tph!

ECA 22's Last Candidate



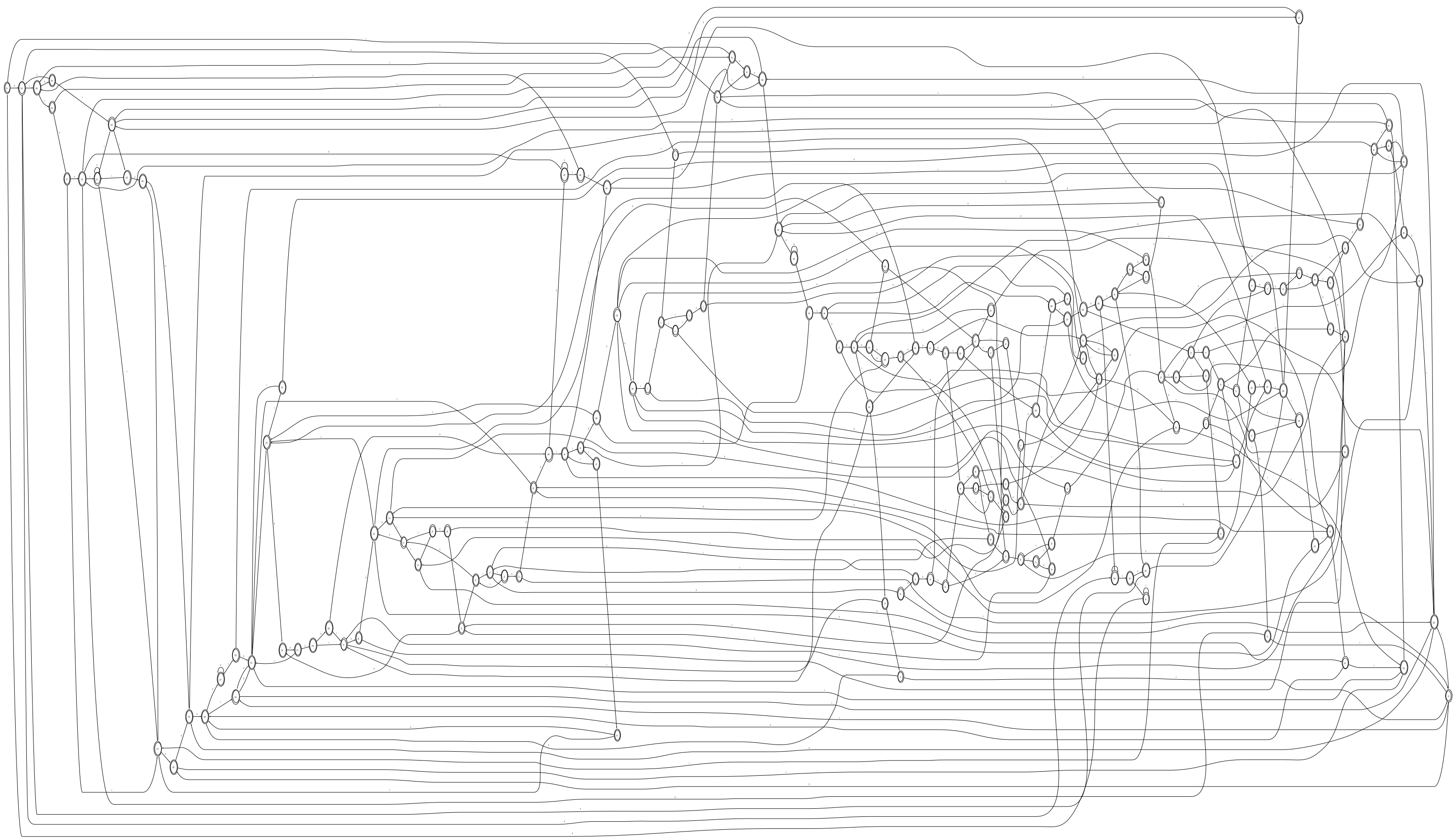
Transducer





Candidate

Iterate of the Last Candidate
161 states and 318 transitions



Theorem: Not a domain.

Proof: Nonexpanding + $h(\Phi(\Lambda)) < h(\Lambda) \Rightarrow$ Candidate contracts.

The Near Term: All Elementary CAs

Push of a Button:

1. CA Pattern Discoverer will automatically build structural theories for all 256 ECAs.

2. Result: a website of about 1000 pages, with several dozen pages per ECA.

Beware of Prophets Expounding the CA Gospel!

The Future: Artificial Science

Implications:

1. Artificial Particle Physics.

2. Automated Pattern Discovery.

3. Theorists unemployed?