Cellular Automata, Decidability and Phasespace

Klaus Sutner

Carnegie Mellon University

NCAM 2009

Outline

Outline

Outline

Outline

Principle of Computational Equivalence

In 2002 Wolfram proposed a vaguely worded Principle of Computational Equivalence (PCE):

> *There are various ways to state the Principle of Computational Equivalence, but probably the most general is just to say that almost all processes that are not obviously simple can be viewed as computations of equivalent sophistication.*

> *PCE . . . has vastly richer implications than the laws of thermodynamics or, for that matter, than essentially any single collection of laws in science.*

The Evidence

A very impressive collection of simulations on various systems such as Turing machine, register machines, tag systems, rewrite systems, combinators, cellular automata.

"Sophisticated" systems can be quite small. Not new, but still: the complexity of many apparently simple systems is surprising.

Nota bene: All systems under consideration here are quite limited in size, one cannot search in any systematic way over spaces of larger systems.

Moreover, the properties in question are undecidable, so there is a lot of heuristics here.

The Evidence

A very impressive collection of simulations on various systems such as Turing machine, register machines, tag systems, rewrite systems, combinators, cellular automata.

"Sophisticated" systems can be quite small. Not new, but still: the complexity of many apparently simple systems is surprising.

Nota bene: All systems under consideration here are quite limited in size, one cannot search in any systematic way over spaces of larger systems.

Moreover, the properties in question are undecidable, so there is a lot of heuristics here.

The Objection

PCE is obviously false: the solution to Post's Problem, due independently to Friedberg and Muchnik in the 1950's, shows that there are recursively enumerable sets of intermediate degree:

$$\emptyset <_T A <_T \emptyset'$$

The "process" of enumerating any such set A is neither trivial nor is it as complicated as, say, the Halting problem.

The Objection

PCE is obviously false: the solution to Post's Problem, due independently to Friedberg and Muchnik in the 1950's, shows that there are recursively enumerable sets of intermediate degree:

$$\emptyset <_T A <_T \emptyset'$$

The "process" of enumerating any such set A is neither trivial nor is it as complicated as, say, the Halting problem.

The Rejoinder

You are cheating. You are constructing a cellular automaton whose behavior has intermediate degree in some technical sense, but underneath it all there is a universal Turing machine. The real computational process is universal.

Wolfram's response (in essence).

Follows a long tradition of uneasiness about intermediate degrees; all known examples are totally artificial: natural r.e. sets that are undecidable are already complete.

The Rejoinder

> *You are cheating. You are constructing a cellular automaton whose behavior has intermediate degree in some technical sense, but underneath it all there is a universal Turing machine. The real computational process is universal.*

Wolfram's response (in essence).

Follows a long tradition of uneasiness about intermediate degrees; all known examples are totally artificial: natural r.e. sets that are undecidable are already complete.

The Rejoinder

> *You are cheating. You are constructing a cellular automaton whose behavior has intermediate degree in some technical sense, but underneath it all there is a universal Turing machine. The real computational process is universal.*

Wolfram's response (in essence).

Follows a long tradition of uneasiness about intermediate degrees; all known examples are totally artificial: natural r.e. sets that are undecidable are already complete.

H. Poincaré

Formerly, when one invented a new function, it was to further some practical purpose; today one invents them in order to make incorrect the reasoning of our fathers, and nothing more will ever be accomplished by these inventions.

A Tough Nut

Wolfram's objection may sound like sour grapes, but there really is a serious problem.

For example, the classical Friedberg-Muchnik construction produces not one but two r.e. sets A and B.

Unfortunately, the disjoint sum $A \oplus B$ is complete.

So we have to hide information to obtain an intermediate set. The whole "process" is indeed complete.

A Tough Nut

Wolfram's objection may sound like sour grapes, but there really is a serious problem.

For example, the classical Friedberg-Muchnik construction produces not one but two r.e. sets A and B.

Unfortunately, the disjoint sum $A \oplus B$ is complete.

So we have to hide information to obtain an intermediate set. The whole "process" is indeed complete.

A Tough Nut

Wolfram's objection may sound like sour grapes, but there really is a serious problem.

For example, the classical Friedberg-Muchnik construction produces not one but two r.e. sets A and B.

Unfortunately, the disjoint sum $A \oplus B$ is complete.

So we have to hide information to obtain an intermediate set. The whole "process" is indeed complete.

A Tough Nut

Wolfram's objection may sound like sour grapes, but there really is a serious problem.

For example, the classical Friedberg-Muchnik construction produces not one but two r.e. sets A and B.

Unfortunately, the disjoint sum $A \oplus B$ is complete.

So we have to hide information to obtain an intermediate set. The whole "process" is indeed complete.

Logic versus Physics

How do we reason reasonably about assertions like PCE?

- Logic, abstract computability theory
  computations independent of representation of data

- Physics, immplementation, concrete computability theory
  computations depend on some particular representation of data

Claim

Discrete dynamics provides a good framework to study the connection, and in particular assertions such as PCE.

Logic versus Physics

How do we reason reasonably about assertions like PCE?

- Logic, abstract computability theory
  computations independent of representation of data

- Physics, immplementation, concrete computability theory
  computations depend on some particular representation of data

Claim

Discrete dynamics provides a good framework to study the connection, and in particular assertions such as PCE.

Logic versus Physics

How do we reason reasonably about assertions like PCE?

- Logic, abstract computability theory
  computations independent of representation of data

- Physics, immplementation, concrete computability theory
  computations depend on some particular representation of data

Claim

Discrete dynamics provides a good framework to study the connection, and in particular assertions such as PCE.

Logic versus Physics

How do we reason reasonably about assertions like PCE?

- Logic, abstract computability theory
  computations independent of representation of data

- Physics, immplementation, concrete computability theory
  computations depend on some particular representation of data

Claim

Discrete dynamics provides a good framework to study the connection, and in particular assertions such as PCE.

Landauer's Principle

In the 1950's John von Neumann speculated that the energy cost associated with manipulating a single bit is at least

$$k\,T\,\ln 2$$

Around 1960 Rolf Landauer took a closer look at the problem and found the following result.

Theorem (Landauer 1961)

Erasing a single bit requires at least $k\,T\,\ln 2$ energy.

Landauer's Principle

In the 1950's John von Neumann speculated that the energy cost associated with manipulating a single bit is at least

$$k\,T\,\ln 2$$

Around 1960 Rolf Landauer took a closer look at the problem and found the following result.

Theorem (Landauer 1961)

Erasing a single bit requires at least $k\,T\,\ln 2$ *energy.*

Rolf Landauer

Information is not a disembodied abstract entity; it is always tied to a physical representation. It is represented by an engraving on a stone tablet, a spin, a charge, a hole in a punched card, a mark on paper, or some other equivalent. This ties the handling of information to all the possibilities and restrictions of our real physical world, its laws of physics and its storehouse of available parts.

David Deutsch

The theory of computation has traditionally been studied almost entirely in the abstract, as a topic in pure mathematics. This is to miss the point of it. Computers are physical objects, and computations are physical processes. What computers can or cannot compute is determined by the laws of physics alone, and not by pure mathematics.

Max Born

About a prototypical physicist:

> *Conscious of the infinite complexities of the phenomena with which he is confronted in every experiment, he resists the idea of considering a theory as something definitive. He therefore abhors the word "Axiom," which in its usual usage evokes the idea of definitive truth.*

Max Born

About a prototypical mathematician:

> *The mathematician, on the contrary, has no business with factual phenomena, but rather with logical interrelations. In Hilbert's language the axiomatic treatment of a discipline implies in no sense a definitive formulation of specific axioms as eternal truths, but rather the following methodological demand: specify the assumptions at the beginning of your deliberation, the stop for a moment and investigate whether or not these assumptions are partly superfluous or contradict each other.*

Hilbert's Sixth Problem

A modest proposal:

Axiomatize all of physics.

Wide open a century later; the Theory-of-Everything remains elusive.

Hilbert's Sixth Problem

A modest proposal:

Axiomatize all of physics.

Wide open a century later; the Theory-of-Everything remains elusive.

Aside: Hypercomputation

It has become fashionable to propose models of computation that supposedly break the "Turing barrier." Models are not interesting, one needs physics-like implementations.

Without an axiomatization of physics this seems somewhat silly.

Aside: Hypercomputation

It has become fashionable to propose models of computation that supposedly break the "Turing barrier." Models are not interesting, one needs physics-like implementations.

Without an axiomatization of physics this seems somewhat silly.

A Sandbox

A Disclaimer: of course, cellular automata will not be the answer to Hilbert's Sixth Problem.

Something like Fredkin's SALT model might come close but probably one needs dynamically evolving topology; even then this is mere speculation at this point.

Question

What is the simplest model that is vaguely physics-like and has interesting computational properties?

A Sandbox

A Disclaimer: of course, cellular automata will not be the answer to Hilbert's Sixth Problem.

Something like Fredkin's SALT model might come close but probably one needs dynamically evolving topology; even then this is mere speculation at this point.

Question

What is the simplest model that is vaguely physics-like and has interesting computational properties?

A Sandbox

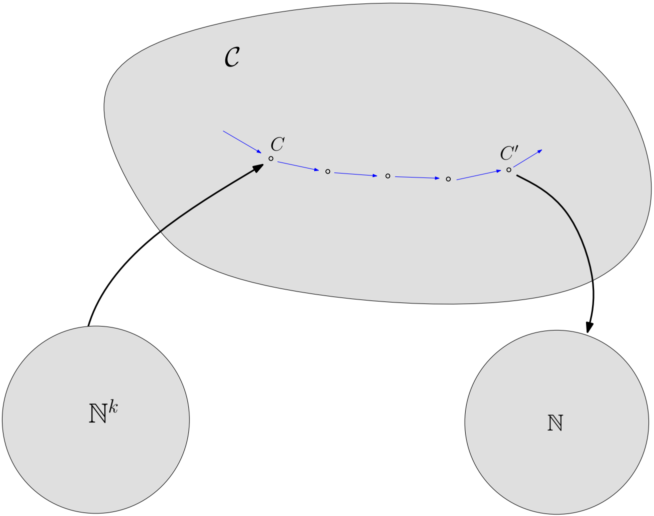
A Disclaimer: of course, cellular automata will not be the answer to Hilbert's Sixth Problem.

Something like Fredkin's SALT model might come close but probably one needs dynamically evolving topology; even then this is mere speculation at this point.

Question

What is the simplest model that is vaguely physics-like and has interesting computational properties?

Phasespace



One-Dimensional Cellular Automata

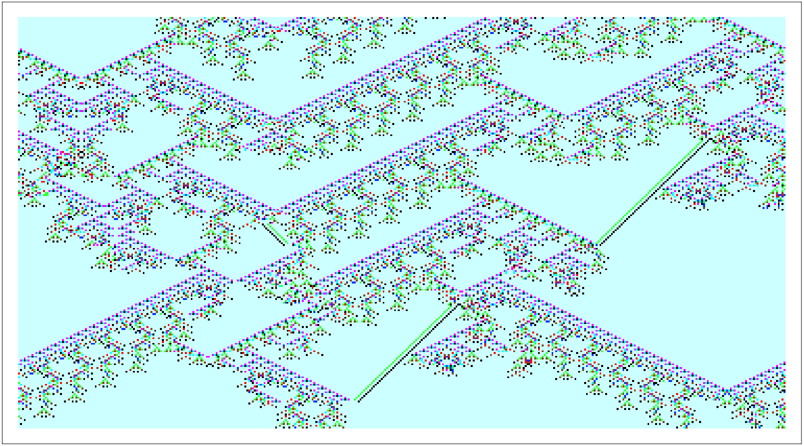
We propose the following model: one-dimensional cellular automata operating on a suitable space of configurations.

Traditional model:

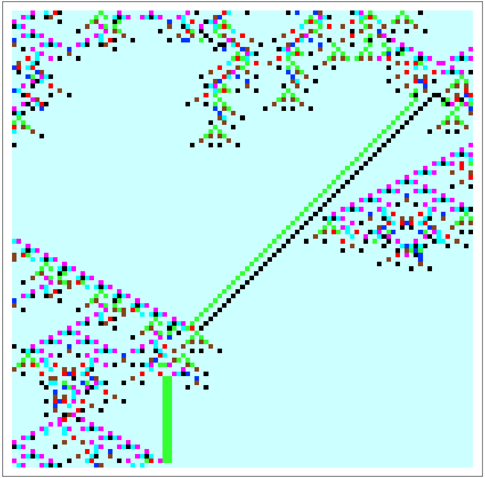
$$\mathfrak{C} = \langle\, \Sigma^{\mathbb{Z}}, \rho \,\rangle$$

where ρ, the global map, is continuous and shift-invariant.

Langton's CA



Zoom



Symbolic Dynamics

Started with Hadamard, Morse, Hedlund.

Cantor space $\Sigma^{\mathbb{Z}}$ has interesting topological properties (compact, zero-dimensional, totally disconnected Hausdorff space), well-studied in classical dynamics.

But, it's ill-behaved when it comes to computation. Even equality becomes problematic.

We need to find a well-behaved subspace of $\Sigma^{\mathbb{Z}}$ that preserves the structure of $\Sigma^{\mathbb{Z}}$ and is computationally amenable.

Symbolic Dynamics

Started with Hadamard, Morse, Hedlund.

Cantor space $\Sigma^{\mathbb{Z}}$ has interesting topological properties (compact, zero-dimensional, totally disconnected Hausdorff space), well-studied in classical dynamics.

But, it's ill-behaved when it comes to computation. Even equality becomes problematic.

We need to find a well-behaved subspace of $\Sigma^{\mathbb{Z}}$ that preserves the structure of $\Sigma^{\mathbb{Z}}$ and is computationally amenable.

Symbolic Dynamics

Started with Hadamard, Morse, Hedlund.

Cantor space $\Sigma^{\mathbb{Z}}$ has interesting topological properties (compact, zero-dimensional, totally disconnected Hausdorff space), well-studied in classical dynamics.

But, it's ill-behaved when it comes to computation. Even equality becomes problematic.

We need to find a well-behaved subspace of $\Sigma^{\mathbb{Z}}$ that preserves the structure of $\Sigma^{\mathbb{Z}}$ and is computationally amenable.

Finite Configurations

Standard cop-out:

$$\mathcal{C} = {}^{\omega}0\,\Sigma^{\star}\,0^{\omega} = \text{ configurations of finite support}$$

Shift invariant and dense, more or less closed under application of ρ.

Violates a basic reflection principle: not closed under ρ^{-1}.

Finite Configurations

Standard cop-out:

$$\mathcal{C} = {}^{\omega}0\,\Sigma^{\star}\,0^{\omega} = \text{ configurations of finite support}$$

Shift invariant and dense, more or less closed under application of ρ.

Violates a basic reflection principle: not closed under ρ^{-1}.

Finite Configurations

Standard cop-out:

$$\mathcal{C} = {}^{\omega}0\,\Sigma^{\star}\,0^{\omega} = \text{ configurations of finite support}$$

Shift invariant and dense, more or less closed under application of ρ.

Violates a basic reflection principle: not closed under ρ^{-1}.

ζ-Regular Languages

We will need a notion of regularity for languages $L \subseteq \Sigma^{\mathbb{Z}}$.

One plausible approach: accepted by a ζ-Büchi automaton.

Definition

A ζ-Büchi automaton $\mathcal{A}$ consists of a transitions system $\langle\, Q, \Sigma, \tau \,\rangle$ and an acceptance condition (I, F) where $I, F \subseteq Q$.

$\mathcal{A}$ accepts $X \in \Sigma^{\mathbb{Z}}$ if there is a run π of $\mathcal{A}$ on X such that $\mathrm{Inf}^-(\pi) \cap I \neq \emptyset$ and $\mathrm{Inf}^+(\pi) \cap F \neq \emptyset$.

Here $\mathrm{Inf}^+(\pi)$ is the forward recurrent set of states and likewise for $\mathrm{Inf}^-(\pi)$.

ζ-Regular Languages

We will need a notion of regularity for languages $L \subseteq \Sigma^{\mathbb{Z}}$.

One plausible approach: accepted by a ζ-Büchi automaton.

Definition

A ζ-Büchi automaton $\mathcal{A}$ consists of a transitions system $\langle Q, \Sigma, \tau \rangle$ and an acceptance condition (I, F) where $I, F \subseteq Q$.

$\mathcal{A}$ accepts $X \in \Sigma^{\mathbb{Z}}$ if there is a run π of $\mathcal{A}$ on X such that $\mathrm{Inf}^{-}(\pi) \cap I \neq \emptyset$ and $\mathrm{Inf}^{+}(\pi) \cap F \neq \emptyset$.

Here $\mathrm{Inf}^{+}(\pi)$ is the forward recurrent set of states and likewise for $\mathrm{Inf}^{-}(\pi)$.

No Coordinates

Mutatis mutandis, very much like ordinary or ω-regular languages, have usual closure properties and algorithmic results.

One small exception: there is no intrinsic coordinate system: there are distinct bi-infinite words that cannot be distinguished by any ζ-Büchi automaton.

- Words are shifts of each other.
- Words have the same cover (i.e., the same set of finite factors) and are recurrent (every finite subword that appears anywhere already appears bi-infinitely often).
- These are the only two possibilities.

No Coordinates

Mutatis mutandis, very much like ordinary or ω-regular languages, have usual closure properties and algorithmic results.

One small exception: there is no intrinsic coordinate system: there are distinct bi-infinite words that cannot be distinguished by any ζ-Büchi automaton.

- Words are shifts of each other.
- Words have the same cover (i.e., the same set of finite factors) and are recurrent (every finite subword that appears anywhere already appears bi-infinitely often).
- These are the only two possibilities.

No Coordinates

Mutatis mutandis, very much like ordinary or ω-regular languages, have usual closure properties and algorithmic results.

One small exception: there is no intrinsic coordinate system: there are distinct bi-infinite words that cannot be distinguished by any ζ-Büchi automaton.

- Words are shifts of each other.
- Words have the same cover (i.e., the same set of finite factors) and are recurrent (every finite subword that appears anywhere already appears bi-infinitely often).
- These are the only two possibilities.

No Coordinates

Mutatis mutandis, very much like ordinary or ω-regular languages, have usual closure properties and algorithmic results.

One small exception: there is no intrinsic coordinate system: there are distinct bi-infinite words that cannot be distinguished by any ζ-Büchi automaton.

- Words are shifts of each other.
- Words have the same cover (i.e., the same set of finite factors) and are recurrent (every finite subword that appears anywhere already appears bi-infinitely often).
- These are the only two possibilities.

No Coordinates

Mutatis mutandis, very much like ordinary or ω-regular languages, have usual closure properties and algorithmic results.

One small exception: there is no intrinsic coordinate system: there are distinct bi-infinite words that cannot be distinguished by any ζ-Büchi automaton.

- Words are shifts of each other.
- Words have the same cover (i.e., the same set of finite factors) and are recurrent (every finite subword that appears anywhere already appears bi-infinitely often).
- These are the only two possibilities.

Decomposition Lemma

One can express ζ-regular languages in terms of ω-regular ones.

Lemma

Every ζ-regular language L has the form

$$L = \bigcup_{i \leq n} U_i^{\mathrm{op}} V_i$$

where all the U_i and V_i are ω-regular languages.

U^{op} means: flip all the strings in U (co-ω words).

Useful to reduce algorithmic problems from ζ to ω.

Automaticity

Consider a relational first order structure

$$\mathfrak{A} = \langle\, A; R_1, R_2, \ldots, R_r \,\rangle$$

$\mathfrak{A}$ is *automatic* if its carrier set and all its relations are all regular.

Originates with Epstein, Holt and Thurston's work on automatic groups; generalized in 1995 Khoussainov and Nerode. Later extended to ω-words and trees.

ζ-Automaticity

Every one-dimensional CA ρ can be associated with a relational structure

$$\mathfrak{C}_\rho = \langle\, \Sigma^{\mathbb{Z}}, \to \,\rangle$$

These structures are obviously ζ-automatic.

Theorem

The first order theory of $\mathfrak{C}_\rho$ is decidable (plus slightly more).

ζ-Automaticity

Every one-dimensional CA ρ can be associated with a relational structure

$$\mathfrak{C}_\rho = \langle\, \Sigma^{\mathbb{Z}}, \to \,\rangle$$

These structures are obviously ζ-automatic.

Theorem

The first order theory of $\mathfrak{C}_\rho$ is decidable (plus slightly more).

FO Definable Relations

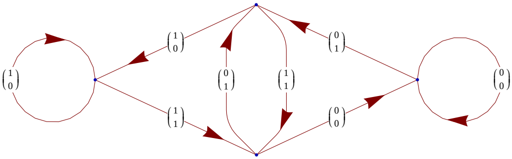
We can effectively convert a formula $\varphi(x_1, x_2, \ldots, x_k)$ into a ζ-Büchi automaton $\mathcal{A}_\varphi$ such that

$$\mathcal{L}(\mathcal{A}_\varphi) = \{\, (x_1, x_2, \ldots, x_k) \in \Sigma^{\mathbb{Z}} \mid \mathfrak{C}_\rho \models \varphi(x_1, x_2, \ldots, x_k) \,\}$$

The basic decision problems for ζ-Büchi automata are decidable (Emptiness, Universality, Inclusion, Equality).

Basic Relation: $x \to y$

...	x_{-3}	x_{-2}	x_{-1}	x_0	x_1	x_2	x_3	...
...	y_{-3}	y_{-2}	y_{-1}	y_0	y_1	y_2	y_3	...



The canonical automaton $\mathcal{A}_\rho(x, y)$ for the local map $\rho(\vec{x}) = x_0 \oplus x_1$.

Induction on Syntax of FO Formula

Read words with k tracks where k is the number of variables.

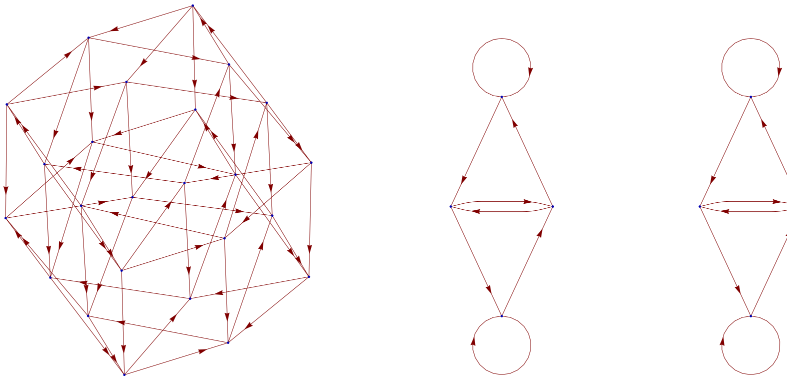
- atomic formulae easy
- $\wedge$ product machine, $\vee$ disjoint union
- $\exists x$ erase track x
- $\neg$ is a fiasco, Safra and worse

Algorithm is not elementary recursive.

Example: Injectivity, Surjectivity

Injectivity and surjectivity are obviously FO properties, so the results from Amoroso-Patt 1972 and KS 1991 follow.

More surprisingly, the quadratic algorithm in the second reference can be obtained from the standard decision procedure plus a bit of tinkering.



Example: Openness

Define $x =_L y$ if $\exists\, n \in \mathbb{Z}\, \forall\, i < n\, (x_i = y_i)$ and $x =_R y$ analogously. Clearly both relations are regular.

But then the global map is open iff it is k-to-1 for some k iff

$$\forall\, x, y, z\, (x \to z \wedge y \to z \wedge (x =_L y \vee x =_R y) \Rightarrow x = y)$$

Thus openess is definable over the extended structure $\mathfrak{C}'_\rho = \langle\, \Sigma^{\mathbb{Z}}, \to, =_L, =_R \,\rangle$ and decidable using the same machinery as for $\mathfrak{C}'_\rho$.

A similar trick helps to speed up surjectivity testing.

Example: Openness

Define $x =_L y$ if $\exists\, n \in \mathbb{Z}\, \forall\, i < n\, (x_i = y_i)$ and $x =_R y$ analogously. Clearly both relations are regular.

But then the global map is open iff it is k-to-1 for some k iff

$$\forall\, x, y, z\, (x \to z \wedge y \to z \wedge (x =_L y \vee x =_R y) \Rightarrow x = y)$$

Thus openess is definable over the extended structure $\mathfrak{C}'_\rho = \langle\, \Sigma^{\mathbb{Z}}, \to, =_L, =_R\, \rangle$ and decidable using the same machinery as for $\mathfrak{C}'_\rho$.

A similar trick helps to speed up surjectivity testing.

Finite CA and Spectra

Finite CA (with some kind of boundary condition) correspond to structures $\mathfrak{C}_\rho^n$ where n is the grid size. Given a sentence φ we want to understand its spectrum:

$$\mathrm{spec}(\varphi) = \{\, n \in \mathbb{N} \mid \mathfrak{C}_\rho^n \models \varphi \,\}$$

Theorem

Spectra are regular: the language $\{\, 0^n \mid n \in \mathrm{spec}(\varphi) \,\}$ *is regular. Moreover, a corresponding finite automaton can be constructed effectively.*

Finite CA and Spectra

Finite CA (with some kind of boundary condition) correspond to structures $\mathfrak{C}_\rho^n$ where n is the grid size. Given a sentence φ we want to understand its spectrum:

$$\mathrm{spec}(\varphi) = \{\, n \in \mathbb{N} \mid \mathfrak{C}_\rho^n \models \varphi \,\}$$

Theorem

Spectra are regular: the language $\{\, 0^n \mid n \in \mathrm{spec}(\varphi) \,\}$ *is regular. Moreover, a corresponding finite automaton can be constructed effectively.*

Example: 3-Cycles

spectrum	# ECA	examples
$\emptyset$	106	$0, 1, 4, 5, 6, \ldots, 251, 252, 253, 254, 255$
$3\mathbb{N}$	108	$2, 3, 10, 11, 16, \ldots, 245, 246, 247, 248, 249$
$5\mathbb{N}$	3	$90, 150, 165$
$6\mathbb{N}$	18	$15, 26, 27, 38, 39, \ldots, 167, 180, 181, 210, 211$
$3, 5 + \mathbb{N}$	4	$62, 118, 131, 145$
$7 + \mathbb{N}$	2	$73, 109$
$9\mathbb{N}$	8	$9, 65, 110, 111, 124, 125, 137, 193$
$7, 9 + \mathbb{N}$	2	$94, 133$
$12\mathbb{N}$	4	$30, 86, 135, 149$
$20\mathbb{N}$	1	105

Almost Periodic Configurations

A good choice for computationally friendly subspace.

$$\begin{aligned}\mathcal{C}_{\mathrm{ap}} &= \{\, {}^{\omega}u\, x\, v^{\omega} \mid u, x, v \in \Sigma^{\star} \,\} \\ &= \text{ almost periodic configurations}\end{aligned}$$

Theorem

For any cellular automaton ρ $\mathfrak{C}_{\rho,\mathsf{ap}}$, the restriction of $\mathfrak{C}_{\rho}$ to $\mathcal{C}_{\mathrm{ap}}$, is an elementary substructure.

Hence the first order theory of $\mathfrak{C}_{\rho,\mathsf{ap}}$ is decidable and coincides with the first order theory of $\mathfrak{C}_{\rho}$.

Almost Periodic Configurations

A good choice for computationally friendly subspace.

$$\begin{aligned} \mathcal{C}_{\mathrm{ap}} &= \{ {}^{\omega} u\, x\, v^{\omega} \mid u, x, v \in \Sigma^{\star} \} \\ &= \text{ almost periodic configurations} \end{aligned}$$

Theorem

For any cellular automaton ρ $\mathfrak{C}_{\rho,\mathsf{ap}}$, the restriction of $\mathfrak{C}_{\rho}$ to $\mathcal{C}_{\mathrm{ap}}$, is an elementary substructure.

Hence the first order theory of $\mathfrak{C}_{\rho,\mathsf{ap}}$ is decidable and coincides with the first order theory of $\mathfrak{C}_{\rho}$.

Orbits

To study computation FOL is too weak, we need orbits:

$$\mathfrak{C}^*_\rho = \langle\, \mathcal{C}, \to, \xrightarrow{*} \,\rangle$$

Theorem

The first order theory of $\mathfrak{C}^*_\rho$ is undecidable in general.

Orbits

To study computation FOL is too weak, we need orbits:

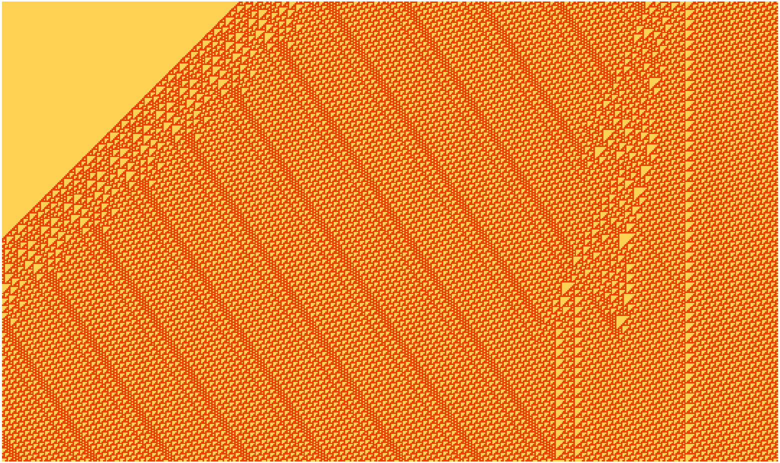
$$\mathfrak{C}^*_\rho = \langle\, \mathcal{C}, \to, \xrightarrow{*} \,\rangle$$

Theorem

The first order theory of $\mathfrak{C}^*_\rho$ is undecidable in general.

ECA 110

As M. Cook has shown, given $\mathcal{C}_{\mathrm{ap}}$ computational universality appears at a surprisingly low level.



Reachability

Problem:	**Reachability Problem**
Instance:	Two almost periodic configuration x and y, a CA ρ.
Question:	Is y in the orbit of x under ρ?

Since almost periodic configurations have a finitary description the problem is clearly r.e.

By Cook's result reachability can be complete even for rather simple CA.

Reachability

Problem:	**Reachability Problem**
Instance:	Two almost periodic configuration x and y, a CA ρ.
Question:	Is y in the orbit of x under ρ?

Since almost periodic configurations have a finitary description the problem is clearly r.e.

By Cook's result reachability can be complete even for rather simple CA.

Reachability

Problem:	**Reachability Problem**
Instance:	Two almost periodic configuration x and y, a CA ρ.
Question:	Is y in the orbit of x under ρ?

Since almost periodic configurations have a finitary description the problem is clearly r.e.

By Cook's result reachability can be complete even for rather simple CA.

Reachability and Degrees

In fact, over $\mathcal{C}_{\mathrm{ap}}$ the situation becomes very complicated.

Theorem (Two-Degree Theorem)

For any two r.e. degrees $\mathbf{d}_1$ and $\mathbf{d}_2$, there is a cellular automaton whose Reachability Problem has degree $\mathbf{d}_1$, and whose Confluence Problem has degree $\mathbf{d}_2$.

Theorem (One-Degree Theorem)

For any r.e. degree $\mathbf{d}$ there is a reversible cellular automaton whose Reachability Problem has degree $\mathbf{d}$.

Reachability and Degrees

In fact, over $\mathcal{C}_{\mathrm{ap}}$ the situation becomes very complicated.

Theorem (Two-Degree Theorem)

For any two r.e. degrees $\mathbf{d}_1$ and $\mathbf{d}_2$, there is a cellular automaton whose Reachability Problem has degree $\mathbf{d}_1$, and whose Confluence Problem has degree $\mathbf{d}_2$.

Theorem (One-Degree Theorem)

For any r.e. degree $\mathbf{d}$ there is a reversible cellular automaton whose Reachability Problem has degree $\mathbf{d}$.

Degree Classification

In fact, over $\mathcal{C}_{\mathrm{ap}}$ we can obtain a rather complicated classification. For any r.e. degree $\mathbf{d}$ let

$$\mathbb{C}_{\mathbf{d}} = \text{ all CA with Reachability of degree } \mathbf{d}$$

All these classes are non-empty, so the semi-lattice of the r.e. degrees is transferred to the CA classification.

Thus, chaos reigns supreme: for example, every countable partial order can be embedded in this classification. Needless to say, it is hard to check membership.

Degree Classification

In fact, over $\mathcal{C}_{\mathrm{ap}}$ we can obtain a rather complicated classification. For any r.e. degree $\mathbf{d}$ let

$$\mathbb{C}_{\mathbf{d}} = \text{ all CA with Reachability of degree } \mathbf{d}$$

All these classes are non-empty, so the semi-lattice of the r.e. degrees is transferred to the CA classification.

Thus, chaos reigns supreme: for example, every countable partial order can be embedded in this classification. Needless to say, it is hard to check membership.

Universality is Hard

The degree classification is higly undecidable.

Theorem

Let $\mathbf{d}$ *be an arbitrary recursively enumerable degree. Then class* $\mathbb{C}_{\mathbf{d}}$ *is* $\Sigma_3^{\mathbf{d}}$*-complete.*

In particular testing for universality (membership in $\mathbb{C}_{\emptyset'}$) is Σ_4-complete.

There is no way in hell to automate Cook's theorem.

Universality is Hard

The degree classification is higly undecidable.

Theorem

Let $\mathbf{d}$ *be an arbitrary recursively enumerable degree. Then class* $\mathbb{C}_{\mathbf{d}}$ *is* $\Sigma_3^{\mathbf{d}}$*-complete.*

In particular testing for universality (membership in $\mathbb{C}_{\emptyset'}$) is Σ_4-complete.

There is no way in hell to automate Cook's theorem.

A Strawman

Here is a plausible definition of a one-dimensional version of a computational process.

π	computor
X_0	initial configuration

Gives rise to the corresponding computation, the orbit of X_0:

$$X_{n+1} = \pi(X_n)$$

Note that a configuration is a snap-shot of the whole computation, not just some selected part.

A Strawman

Here is a plausible definition of a one-dimensional version of a computational process.

π	computor
X_0	initial configuration

Gives rise to the corresponding computation, the orbit of X_0:

$$X_{n+1} = \pi(X_n)$$

Note that a configuration is a snap-shot of the whole computation, not just some selected part.

The Observer

How do we extract results from a computation?

τ observer

Given a computational process $\langle \pi, X_0 \rangle$ and an observer τ the corresponding output or observation as

$$O_\rho = \{ \tau(X_n) \mid n \geq 0 \}$$

The Observer

How do we extract results from a computation?

τ observer

Given a computational process $\langle \pi, X_0 \rangle$ and an observer τ the corresponding output or observation as

$$O_\rho = \{\, \tau(X_n) \mid n \geq 0 \,\}$$

Allocating Blame

Constraints on the complexity of ρ and τ:

ρ	a one-dim CA over $\mathcal{C}_{\mathrm{ap}}$
X_0	almost periodic configuration
τ	constant space word map

The computor and observer can only do a little word processing.

Notably, it's not the observer's fault if the output O_τ is complete.

Allocating Blame

Constraints on the complexity of ρ and τ:

ρ	a one-dim CA over $\mathcal{C}_{\mathrm{ap}}$
X_0	almost periodic configuration
τ	constant space word map

The computor and observer can only do a little word processing.

Notably, it's not the observer's fault if the output O_τ is complete.

Allocating Blame

Constraints on the complexity of ρ and τ:

ρ	a one-dim CA over $\mathcal{C}_{\mathrm{ap}}$
X_0	almost periodic configuration
τ	constant space word map

The computor and observer can only do a little word processing.

Notably, it's not the observer's fault if the output O_τ is complete.

Intermediate Processes

So the observer uncovers some part of the computation but does not increase the complexity. Moreover, she may hide some details.

Definition

A computational process is

- undecidable if there is some observer τ such that O_τ is undecidable.
- universal if there is some observer τ such that O_τ is r.e.-complete.
- intermediate if it is undecidable but fails to be universal.

Intermediate Processes

So the observer uncovers some part of the computation but does not increase the complexity. Moreover, she may hide some details.

Definition

A computational process is

- undecidable if there is some observer τ such that O_τ is undecidable.
- universal if there is some observer τ such that O_τ is r.e.-complete.
- intermediate if it is undecidable but fails to be universal.

Anti Climax

Burning Question: Is there an intermediate computational process?

Weak Answer: All the known constructions of intermediate degrees seem to produce commplete processes.

More or less only 2.5 choices

- Friedberg-Muchnik priority construction
- Daley Busy Beaver priority construction
- Kucera priority-free construction

Conjecture

Conspicuous absence thereof.

Anti Climax

Burning Question: Is there an intermediate computational process?

Weak Answer: All the known constructions of intermediate degrees seem to produce commplete processes.

More or less only 2.5 choices

- Friedberg-Muchnik priority construction
- Daley Busy Beaver priority construction
- Kucera priority-free construction

Conjecture

Conspicuous absence thereof.

Anti Climax

Burning Question: Is there an intermediate computational process?

Weak Answer: All the known constructions of intermediate degrees seem to produce commplete processes.

More or less only 2.5 choices

- Friedberg-Muchnik priority construction
- Daley Busy Beaver priority construction
- Kucera priority-free construction

Conjecture

Conspicuous absence thereof.

Anti Climax

Burning Question: Is there an intermediate computational process?

Weak Answer: All the known constructions of intermediate degrees seem to produce commplete processes.

More or less only 2.5 choices

- Friedberg-Muchnik priority construction
- Daley Busy Beaver priority construction
- Kucera priority-free construction

Conjecture

Conspicuous absence thereof.

- Computation and Physics
- ζ-Automaticity
- Computation and CA

4 Open Problems

Algorithmic Questions

- What fragment for FOL on $\mathfrak{C}_\rho$ can be decided efficiently?
- Can one exploit the structure of ζ-words to get better algorithms (speed up Safra, Kupferman's ranking method, etc.)?
- How about simple CA such as linear rules?
- Is there an interesting class of CA for which stronger logics can be handled?

Understanding the Structures $\mathfrak{C}_\rho$

General cookbook:

- Similarity: when are structures isomorphic, elementarily equivalent.
- Definability: what is definable in the structure; is the class axiomatizable.
- Decidability: is $\mathcal{A} \models \varphi$ decidable, is the elementary/first-order theory of $\mathcal{A}$ decidable.
- Connections: what are relationships between these and to other structures.
- Logics: how about other logics than first-order.

Aside: is "cellularity" of regular languages decidable?

Understanding the Structures $\mathfrak{C}_\rho$

General cookbook:

- Similarity: when are structures isomorphic, elementarily equivalent.
- Definability: what is definable in the structure; is the class axiomatizable.
- Decidability: is $\mathcal{A} \models \varphi$ decidable, is the elementary/first-order theory of $\mathcal{A}$ decidable.
- Connections: what are relationships between these and to other structures.
- Logics: how about other logics than first-order.

Aside: is “cellularity” of regular languages decidable?

FO Theories

- Is $\mathrm{Th}(\mathfrak{C}_\rho)$ a useful measure of complexity?
- How about $\mathrm{Th}(\mathfrak{C}_\rho^*)$?
- Can these theories produce an interesting classification?
- Is the theory of Wolfram Class III the same as Class IV?
- What happens with stronger logics?

Principle of Computational Equivalence

- Does PCE hold for elementary CA?
- What is the least radius/number of symbols where it fails?
- How about reversible cellular automata?
- Are all known constructions of intermediate degrees complete as CPs?
- Is there an intermediate CP?

Thank You