Signal Machines: Euclidean dynamical system

Introduction and universalities

Florent Becker¹, Jérôme Durand-Lose¹
Mohammad-Hadi Foroughmand-Araabi² and Sama Goliaei³

¹ Laboratoire d’Informatique Fondamentale d’Orléans, Université d’Orléans, France
² Sharif University of Technology, Tehran, Iran
³ University of Tehran, Tehran, Iran

University of South Florida (USF) — May 25th 2017
1 Introduction to Signal Machines
   • Definition
   • Fractals
   • Computing (Turing-) Universality

2 Intrinsic Universality
   • Concept and Definition
   • Global Scheme
   • Shrink and Test
   • Macro-Collision

3 Conclusion
1 Introduction to Signal Machines
- Definition
- Fractals
- Computing (Turing-) Universality

2 Intrinsic Universality
- Concept and Definition
- Global Scheme
- Shrink and Test
- Macro-Collision

3 Conclusion
Self Completing Drawing

2D Euclidean Space
- color line segments
- orientation (not going back)
Self Completing Drawing

2D Euclidean Space
- color line segments
- orientation (not going back)
- Potential enlargement
- Intersection

\[ \{b, r\} \rightarrow \{g\} \]

Direction/Slope Imposed by the Color

Origin of the model
Self Completing Drawing

2D Euclidean Space
- color line segments
- orientation (not going back)

Potential enlargement
- Intersection
- Extension

Rewriting/Collision rule:
\{b, r\} \rightarrow \{g\}

Direction/Slope Imposed by the Color
easier

Origin of the model
Self Completing Drawing

2D Euclidean Space
- color line segments
- orientation (not going back)

Potential enlargement
- Intersection
- Extension

Rewriting/Collision rule
- \( \{b, r\} \rightarrow \{g\} \)
Self Completing Drawing

2D Euclidean Space
- color line segments
- orientation (not going back)

Potential enlargement
- Intersection
- Extension

Rewriting/Collision rule
- \{b, r\} → \{g\}

Direction/Slope Imposed by the Color
- (easier)
- origin of the model
Cellular Automata
Cellular Automata
Cellular Automata
Cellular Automata
Cellular Automata: Signal Use

Firing Quad Synchronization [Goto, 1966]
CA: Signal Design

Generation of Primes [Fischer, 1965]

![Diagram](image-url)
CA: Signal Analyzing

(Evolutionary generated) Automatic blinking [Das et al., 1995]

(a) Space-time diagram. (b) Filtered space-time diagram.

Figure 1: (a) Space-time diagram of sync starting with a random initial condition. (b) The same space-time diagram after filtering with a spatial transducer that maps all domains to white and all defects to black. Greek letters label particles described in the text.

Figure 1a gives a space-time diagram for one of the GA-discovered CAs with 100% performance, here called sync. This diagram plots 75 successive configurations on a lattice of size N = 75 (with time going down the page) starting from a randomly chosen IC, with 1-sites colored black and 0-sites colored white. In this example, global synchronization occurs at time step 58.

How are we to understand the strategy employed by sync to reach global synchronization?

Notice that, under the GA, while crossover and mutation act on the local mappings comprising a chromosome. This defines one generation of the GA; it is repeated G times for one GA run. 

\( F_I (\gamma / \gamma) \) is a random variable since its value depends on the particular set of ICs selected to evaluate. Thus, a CA’s fitness varies stochastically from generation to generation. For this reason, we choose a new set of ICs at each generation.

For our experiments, we set \( P_I = 1/0/0/, E_I = 2/0/; I_I = 1/0/0/, m_I = 2/; and G_I = 5/0/. \)

M was chosen from a Poisson distribution with mean 320 (slightly greater than 2\( N \)). Varying M prevents selecting CAs that are adapted to a particular M. A justification of these parameter settings is given in [9].

We performed a total of 65 GA runs. Since \( F_I (1/0/0/) \) is only a rough estimate of performance, we more stringently measured the quality of the GA’s solutions by calculating \( P_{\gamma} (1/0/4/) \) with \( N = 1/4/9/; 5/9/9/; 9/9/9/; \) for the best CAs in the final generation of each run. In 20% of the runs, the GA discovered successful CAs (\( P_{\gamma} (1/0/4/) = 1 : 0/) \). More detailed analysis of these successful CAs showed that although they were distinct in detail, they used similar strategies for performing the synchronization task. Interestingly, when the GA was restricted to evolve CAs with \( r = 1 \) and \( r = 2 \), all the evolved CAs had \( P_{\gamma} (1/0/4/) \) for \( N = 1/4/9/; 5/9/9/; 9/9/9/; \).

(Better performing CAs with \( r = 2 \) can be designed by hand.) Thus, \( r = 3 \) appears to be the minimal radius for which the GA can successfully solve this problem.
**Signals**

- Signal (meta-signal)
- Collision (rule)
1. **Introduction to Signal Machines**
   - Definition
   - Fractals
   - Computing (Turing-) Universality

2. **Intrinsic Universality**
   - Concept and Definition
   - Global Scheme
   - Shrink and Test
   - Macro-Collision

3. **Conclusion**
Vocabulary and Example: Find the Middle

Meta-signals (speed)

\[ M \quad (0) \]

Collision rules
Vocabulary and Example: Find the Middle

Meta-signals (speed)

| M (0) |
| div (3) |

Collision rules

| div M M |
| M |
| M |
| M |
Vocabulary and Example: Find the Middle

**Meta-signals (speed)**

- $M$ (0)
- div (3)
- hi (1)
- lo (3)

**Collision rules**

$$\{ \text{div, } M \} \rightarrow \{ M, \text{hi, lo} \}$$
Vocabulary and Example: Find the Middle

Meta-signals (speed)

- M (0)
- div (3)
- hi (1)
- lo (3)
- back (-3)

Collision rules

\[
\begin{align*}
\{ \text{div, } M \} &\rightarrow \{ M, \text{hi, lo} \} \\
\{ \text{lo, } M \} &\rightarrow \{ \text{back, } M \}
\end{align*}
\]
Vocabulary and Example: Find the Middle

Meta-signals (speed)

\[
\begin{align*}
M & \quad (0) \\
\text{div} & \quad (3) \\
\text{hi} & \quad (1) \\
\text{lo} & \quad (3) \\
\text{back} & \quad (-3)
\end{align*}
\]

Collision rules

\[
\begin{align*}
\{ \text{div, M} \} & \rightarrow \{ \text{M, hi, lo} \} \\
\{ \text{lo, M} \} & \rightarrow \{ \text{back, M} \} \\
\{ \text{hi, back} \} & \rightarrow \{ \text{M} \}
\end{align*}
\]
Another Example

<table>
<thead>
<tr>
<th>Speed</th>
</tr>
</thead>
</table>
| $\mu_1$ | 1  
| $\mu_2$ | $-1/2$  
| $\mu_3$ | 3  
| $\mu_4$ | 0  

Collision rules

- $\{ \mu_1, \mu_2 \} \rightarrow \{ \mu_2, \mu_1, \mu_3 \}$
- $\{ \mu_3, \mu_4 \} \rightarrow \{ \mu_2 \}$
- $\{ \mu_4, \mu_2 \} \rightarrow \{ \mu_2, \mu_4 \}$
Complex Dynamics
Complex Dynamics
1 Introduction to Signal Machines
   - Definition
   - Fractals
   - Computing (Turing-) Universality

2 Intrinsic Universality
   - Concept and Definition
   - Global Scheme
   - Shrink and Test
   - Macro-Collision

3 Conclusion
Examples
Cantor of any Hausdorff Dimension [Senot, 2013]
Second Order
1 Introduction to Signal Machines
- Definition
- Fractals
- Computing (Turing-) Universality

2 Intrinsic Universality
- Concept and Definition
- Global Scheme
- Shrink and Test
- Macro-Collision

3 Conclusion
Adding

\[
\begin{array}{c}
1^* \\
0 \\
0 \\
1 \\
0 \\
1 \\
\end{array} + \begin{array}{c}
1 \\
1 \\
0 \\
1 \\
1 \\
0 \\
\end{array} = \begin{array}{c}
1^* \\
0 \\
0 \\
1 \\
0 \\
1 \\
\end{array}
\]

MSB 1* 0 0 1 0 LSB

MSB 1* 1 0 1 LSB

MSB 1* 0 0 1 0 LSB
(Turing)-Computing

**Turing Machine**

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

**Simulation**

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```

```
q_f
^ b b a b #
```
Signal Machines: Euclidean dynamical system

Introduction to Signal Machines

Computing (Turing-) Universality

(Turing)-Computing

Simulation

Rationnall Machine

- speeds $\in \mathbb{Q}$
- initial positions $\in \mathbb{Q}$
- $\Rightarrow$ coordinates of any collision $\in \mathbb{Q}$
- exact computation on a computer/TM

Undecidability

- finite number de collisions
- meta-signal appereance
- use of a rule
- disappearing of all signals
- involvement of a signal in any collision
- extension on the side, etc.
1 Introduction to Signal Machines
- Definition
- Fractals
- Computing (Turing-) Universality

2 Intrinsic Universality
- Concept and Definition
- Global Scheme
- Shrink and Test
- Macro-Collision

3 Conclusion
1. Introduction to Signal Machines
   - Definition
   - Fractals
   - Computing (Turing-) Universality

2. Intrinsic Universality
   - Concept and Definition
   - Global Scheme
   - Shrink and Test
   - Macro-Collision

3. Conclusion
Signal Machines: Euclidean dynamical system

Intrinsic Universality

Concept and Definition

Concept
- to represent all others
- capability of any/all
- most general (universal)

Examples
- micro-processor, FPGA, JVM
- Java, C, Php
- Turing machine + Church-Turing Thesis \(\sim \) computability theory
For dynamical systems

**Intrinsic Universality**

Being able to *simulate* any other dynamical system of its class.

**Cellular Automata**

- reversible [Durand-Lose, 1997]

**Tile Assembly Systems**

- possible at T=2 and above [Woods, 2013]
- impossible at T=1 [Meunier et al., 2014]
Simulation for Signal Machines

Space-Time Diagram Mimicking

Signal Machine Simulation

$U_S$ simulates $M$ if there is function from the configurations of $M$ to the ones of $U_S$ s.t. the space-time issued from the image always mimics the original one.
Our result [Submitted]

**Theorem**

For any finite set of real numbers $S$, there is a signal machine $U_S$, that can simulate any machine whose speeds belong to $S$.

**Theorem**

The set of $U_S$ where $S$ ranges over finite sets of real numbers is an intrinsically universal family of signal machines.

**Rest of the talk**

Let $S$ be any finite set of real numbers, let $\mathcal{M}$ be any signal machine whose speeds belongs to $S$, $U_S$ is progressively constructed as simulation is presented.
1 Introduction to Signal Machines
   - Definition
   - Fractals
   - Computing (Turing-) Universality

2 Intrinsic Universality
   - Concept and Definition
   - Global Scheme
   - Shrink and Test
   - Macro-Collision

3 Conclusion
Macro-Signal

- Meta-signal of $\mathcal{M}$ identified with numbers
- Unary encoding of numbers

Macro-Signal Structure

$i\mu^k$: $k$th signal, $i$th speed

Collision Rules encoding

support zone
Global scheme

When Support Zones Meet (rough vision)

- Start the macro-collision (if applicable)
Global scheme

When Support Zones Meet (rough vision)

- Start the macro-collision (if applicable)

When Support Zones Meet

1. provide a delay
2. test if macro-collision is appropriate and what macro-signals are involved
3. if OK
   - start the macro-collision
1 Introduction to Signal Machines
   - Definition
   - Fractals
   - Computing (Turing-) Universality

2 Intrinsic Universality
   - Concept and Definition
   - Global Scheme
   - Shrink and Test
   - Macro-Collision

3 Conclusion
Signal Machines: Euclidean dynamical system
Intrinsic Universality
Shrink and Test

**Good (?) Cases**

**Bad Case**
Whole Preparation (cropped on both side)
Signal Machines: Euclidean dynamical system
Intrinsic Universality
Shrink and Test

Shrinking Unit
Shrink
Testing for Other main Signals
Detecting Potential Overlaps
1 Introduction to Signal Machines
   • Definition
   • Fractals
   • Computing (Turing-) Universality

2 Intrinsic Universality
   • Concept and Definition
   • Global Scheme
   • Shrink and Test
   • Macro-Collision

3 Conclusion
Removing Unused Tables and Sending ids to Table
Collision Rules Encoding

One rule after the other

Encoding of \( \{ 3\mu^1, 7\mu^4, 8\mu^5 \} \rightarrow \{ 2\mu^3, 4\mu^1 \} \) in the direction \( i \).
Comparison of id’s in the if-part of a Rule
Rule Selection
Generating the Output
1 Introduction to Signal Machines
   - Definition
   - Fractals
   - Computing (Turing-) Universality

2 Intrinsic Universality
   - Concept and Definition
   - Global Scheme
   - Shrink and Test
   - Macro-Collision

3 Conclusion
All Together

Signal Machines

- Rich world

Theorems are proved
Open problems

- single intrinsically universal signal machine (with amended simulation definition)
- discretization into CA (Tom BESSON’s Theses) into Tile Assembly System
- robustness
- complexity (non-det. signal machines), ordinal clocking

References

- Visual introduction (not much)
  

- Articles by JDL can be accessed at
  


Conclusion

Intrinsic Universality in Tile Self-Assembly Requires Cooperation.

Two-States Bilinear Intrinsically Universal Cellular Automata.
In FCT ’01, number 2138 in LNCS, pages 369–399. Springer.

Modèle géométrique de calcul : fractales et barrières de complexité.
Thèse de doctorat, Université d’Orléans.

Intrinsic Universality and the Computational Power of Self-Assembly.