Expressive Power of Recurrent Neural Networks over Infinite Input Streams

Jérémie Cabessa
Joint work with Olivier Finkel

Laboratoire d'Économie Mathématique
Université Paris 2

13 April 2017

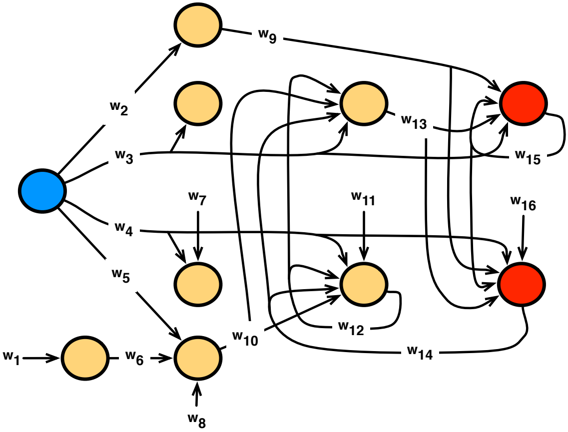
Introduction

- The computational capabilities of recurrent neural networks have mainly been studied in the context of classical computation: McCulloch & Pitts, Turing, Kleene, von Neumann, Minsky, Papert,. . . , Siegelmann & Sontag,. . .
- We provide a characterization of the computational power of recurrent neural networks in terms of their attractor dynamics, i.e., in the context of infinite input stream computation.

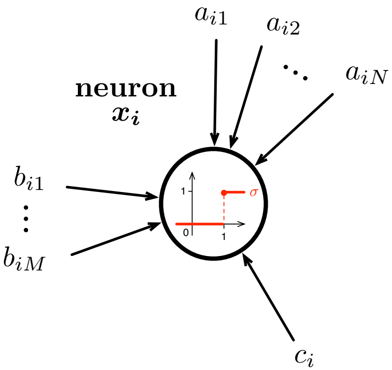
Introduction

- The computational capabilities of recurrent neural networks have mainly been studied in the context of classical computation: McCulloch & Pitts, Turing, Kleene, von Neumann, Minsky, Papert,. . . , Siegelmann & Sontag,. . .
- We provide a characterization of the computational power of recurrent neural networks in terms of their attractor dynamics, i.e., in the context of infinite input stream computation.

Recurrent Neural Network

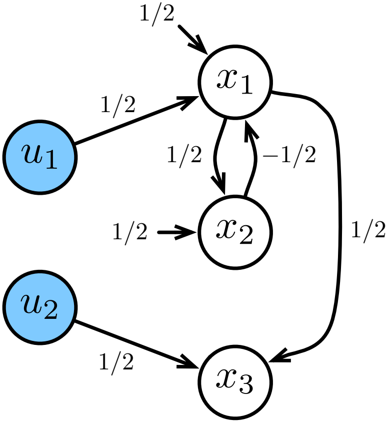


Boolean Recurrent Neural Networks

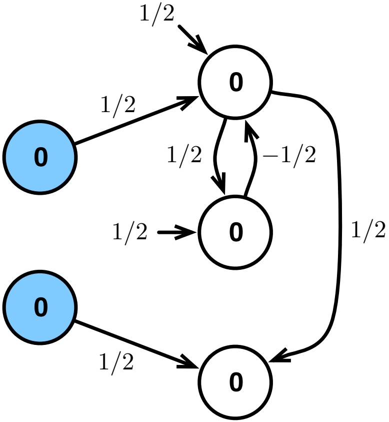


$$x_i(t+1) = \theta\left(\sum_{j=1}^{N} a_{ij} \cdot x_j(t) + \sum_{j=1}^{M} b_{ij} \cdot u_j(t) + c_i\right)$$

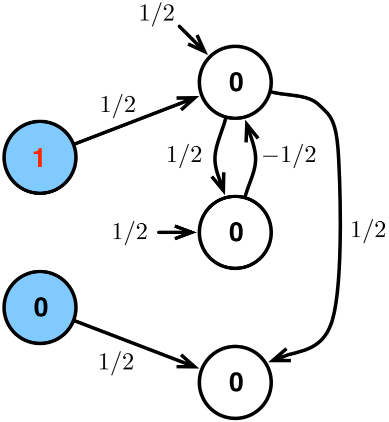
Boolean Recurrent Neural Networks



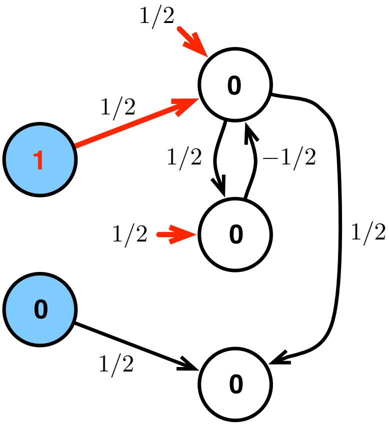
Boolean Recurrent Neural Networks



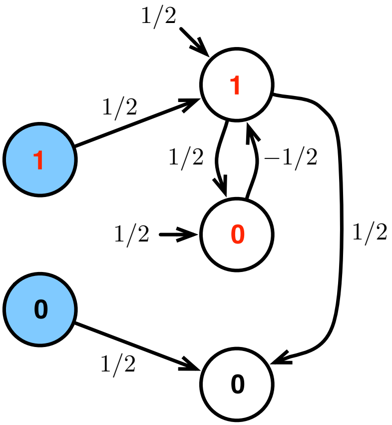
Boolean Recurrent Neural Networks



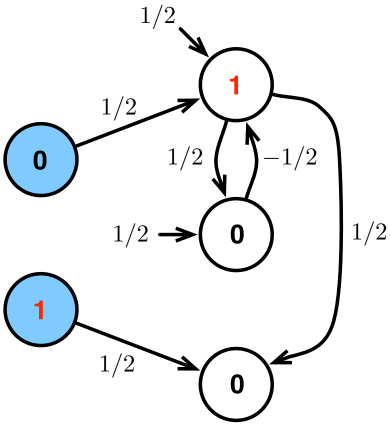
Boolean Recurrent Neural Networks



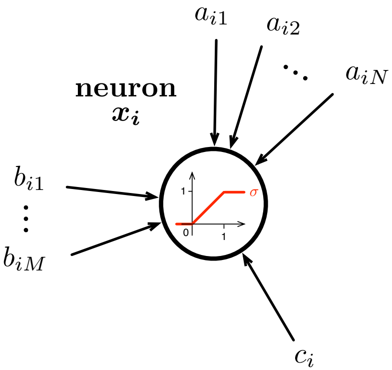
Boolean Recurrent Neural Networks



Boolean Recurrent Neural Networks

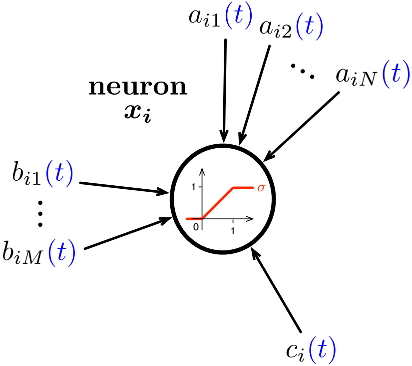


Sigmoidal Recurrent Neural Networks



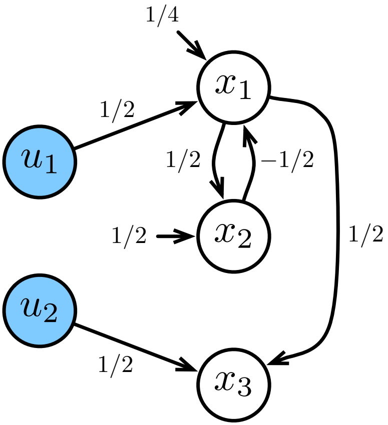
$$x_i(t+1) = \sigma\left(\sum_{j=1}^{N} a_{ij} \cdot x_j(t) + \sum_{j=1}^{M} b_{ij} \cdot u_j(t) + c_i\right)$$

Sigmoidal Recurrent Neural Networks

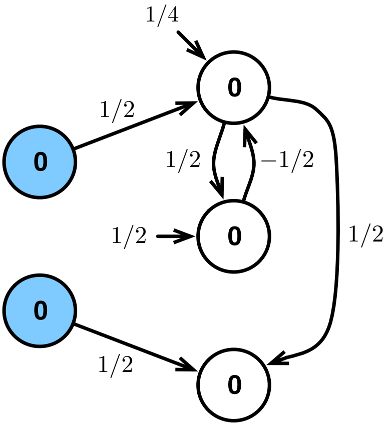


$$x_i(t+1) = \sigma\left(\sum_{j=1}^{N} a_{ij}(t) \cdot x_j(t) + \sum_{j=1}^{M} b_{ij}(t) \cdot u_j(t) + c_i(t)\right)$$

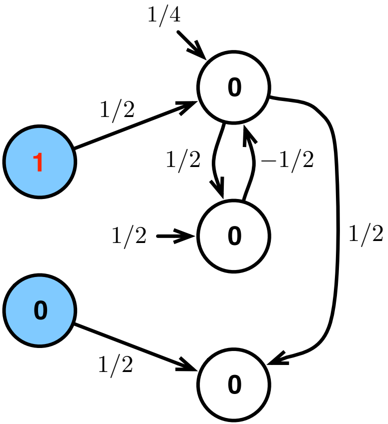
Sigmoidal Recurrent Neural Networks



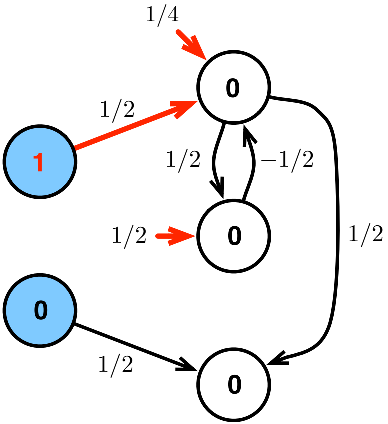
Sigmoidal Recurrent Neural Networks



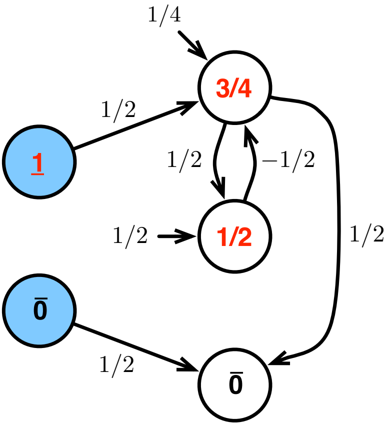
Sigmoidal Recurrent Neural Networks



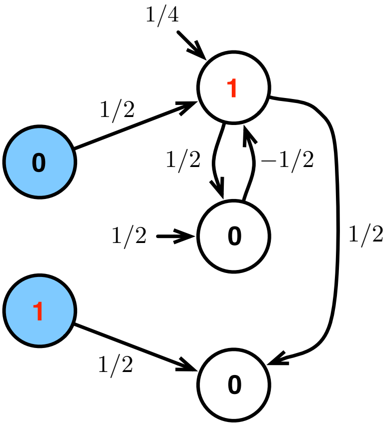
Sigmoidal Recurrent Neural Networks



Sigmoidal Recurrent Neural Networks



Sigmoidal Recurrent Neural Networks



Recurrent Neural Networks

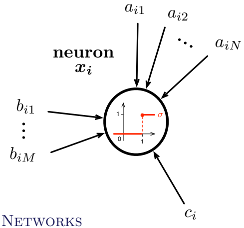
We consider eight models of RNNs:

1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev$_2$-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev$_2$-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s

Recurrent Neural Networks

We consider eight models of RNNs:

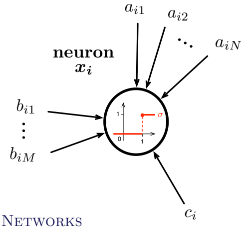
1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev$_2$-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev$_2$-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s



Recurrent Neural Networks

We consider eight models of RNNs:

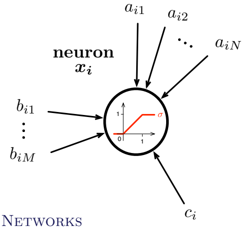
1. Boolean rational RNNs: B-RNN[ℚ]s
2. Boolean real RNNs: B-RNN[ℝ]s
3. Sigmoidal static rational RNNs: St-RNN[ℚ]s
4. Sigmoidal static real RNNs: St-RNN[ℝ]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev_2-RNN[ℚ]s
6. Sigmoidal bi-valued evolving real RNNs: Ev_2-RNN[ℝ]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[ℚ]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[ℝ]s



Recurrent Neural Networks

We consider eight models of RNNs:

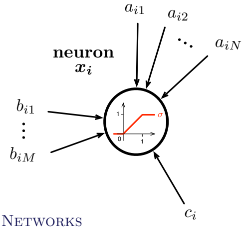
1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev$_2$-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev$_2$-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s



Recurrent Neural Networks

We consider eight models of RNNs:

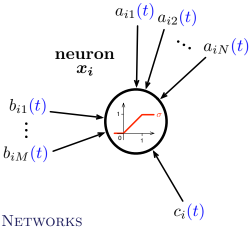
1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev_2-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev_2-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s



Recurrent Neural Networks

We consider eight models of RNNs:

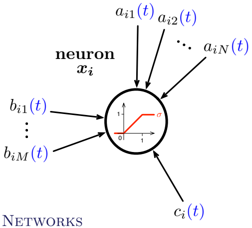
1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev$_2$-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev$_2$-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s



Recurrent Neural Networks

We consider eight models of RNNs:

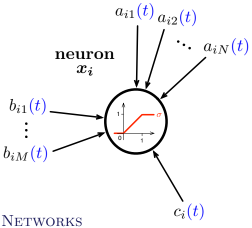
1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev$_2$-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev$_2$-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s



Recurrent Neural Networks

We consider eight models of RNNs:

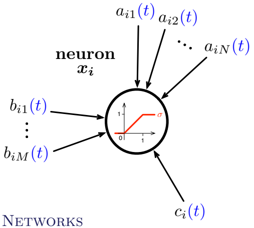
1. Boolean rational RNNs: B-RNN[$\mathbb{Q}$]s
2. Boolean real RNNs: B-RNN[$\mathbb{R}$]s
3. Sigmoidal static rational RNNs: St-RNN[$\mathbb{Q}$]s
4. Sigmoidal static real RNNs: St-RNN[$\mathbb{R}$]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev_2-RNN[$\mathbb{Q}$]s
6. Sigmoidal bi-valued evolving real RNNs: Ev_2-RNN[$\mathbb{R}$]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[$\mathbb{Q}$]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[$\mathbb{R}$]s



Recurrent Neural Networks

We consider eight models of RNNs:

1. Boolean rational RNNs: B-RNN[ℚ]s
2. Boolean real RNNs: B-RNN[ℝ]s
3. Sigmoidal static rational RNNs: St-RNN[ℚ]s
4. Sigmoidal static real RNNs: St-RNN[ℝ]s
5. Sigmoidal bi-valued evolving rational RNNs: Ev_2-RNN[ℚ]s
6. Sigmoidal bi-valued evolving real RNNs: Ev_2-RNN[ℝ]s
7. Sigmoidal general evolving rational RNNs: Ev-RNN[ℚ]s
8. Sigmoidal general evolving real N-RNNs: Ev-RNN[ℝ]s

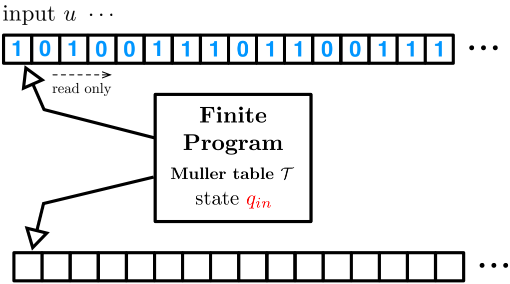


Results (Classical Computation)

	Boolean	Static	Bi-valued Evolving	Evolving
$\mathbb{Q}$	FSA	TM	TM/poly(A)	TM/poly(A)
	REG	**P**	**P/poly**	**P/poly**
	Kl 56, Mi 67	Si & So 95	Ca & Si 11,14	Ca & Si 11,14
$\mathbb{R}$	FSA	TM/poly(A)	TM/poly(A)	TM/poly(A)
	REG	**P/poly**	**P/poly**	**P/poly**
	Kl 56, Mi 67	Si & So 94	Ca & Si 11,14	Ca & Si 11,14

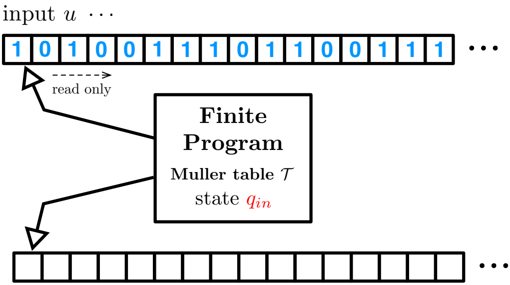
Muller Turing Machine

A *Muller Turing machine* consists of a classical TM with Muller acceptance condition. Muller table $\mathcal{T}$: collection of sets of states.



Muller Turing Machine

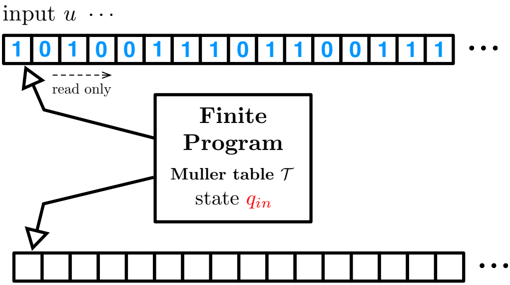
A *Muller Turing machine* consists of a classical TM with Muller acceptance condition. Muller table $\mathcal{T}$: collection of sets of states.



- The ω-word u is *accepted* by $\mathcal{M}$ if $\inf(\rho_u) \in \mathcal{T}$.

Muller Turing Machine

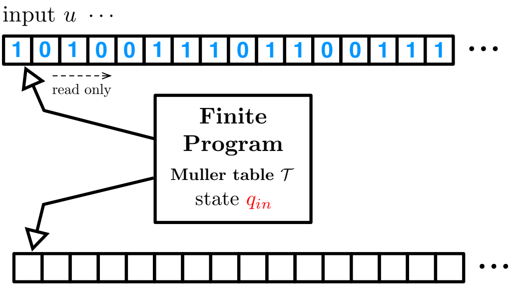
A *Muller Turing machine* consists of a classical TM with Muller acceptance condition. Muller table $\mathcal{T}$: collection of sets of states.



- The ω-word u is *rejected* by $\mathcal{M}$ if $\inf(\rho_u) \notin \mathcal{T}$.

Muller Turing Machine

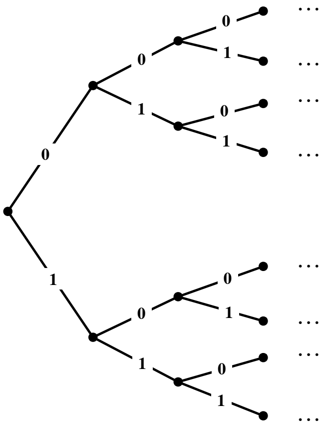
A *Muller Turing machine* consists of a classical TM with Muller acceptance condition. Muller table $\mathcal{T}$: collection of sets of states.



- Set of ω-words accepted is the ω-language *recognized* by $\mathcal{M}$.

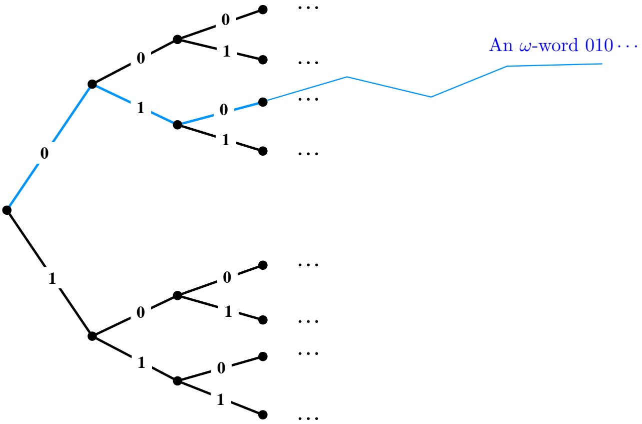
Topology

The Cantor space $\{0,1\}^\omega$
the set of infinite sequences of bits



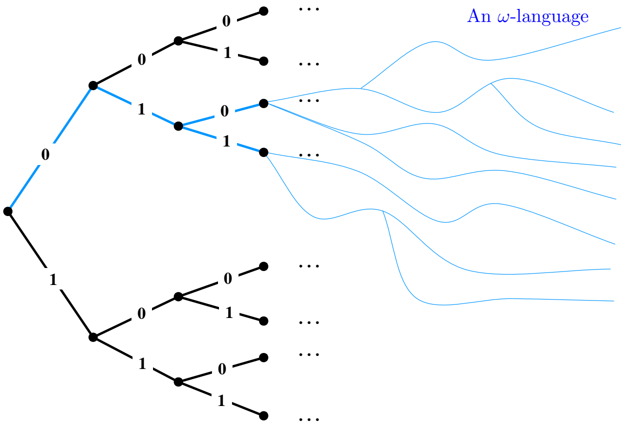
Topology

The Cantor space $\{0,1\}^\omega$
the set of infinite sequences of bits



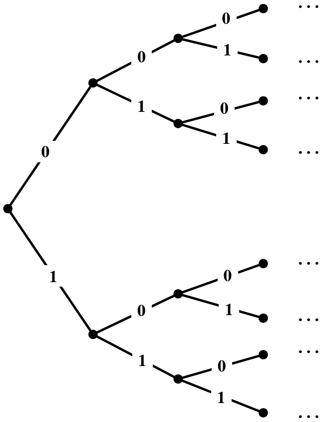
Topology

The Cantor space $\{0,1\}^\omega$
the set of infinite sequences of bits



Topology

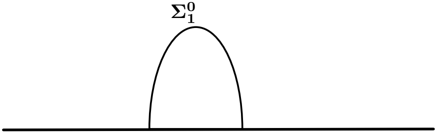
The Cantor space $\{0,1\}^\omega$
the set of infinite sequences of bits



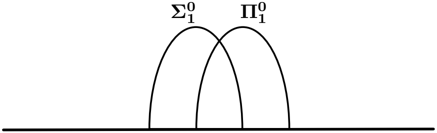
equipped with product topology
of the discrete topology on $\{0,1\}$

Topology

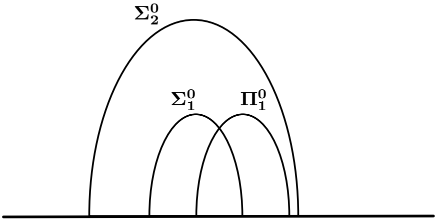
Topology



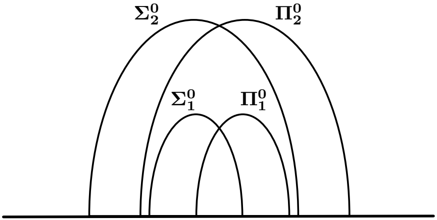
Topology



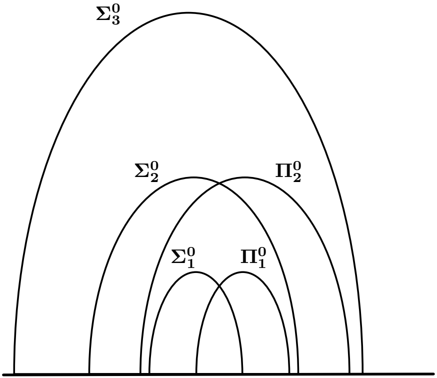
Topology



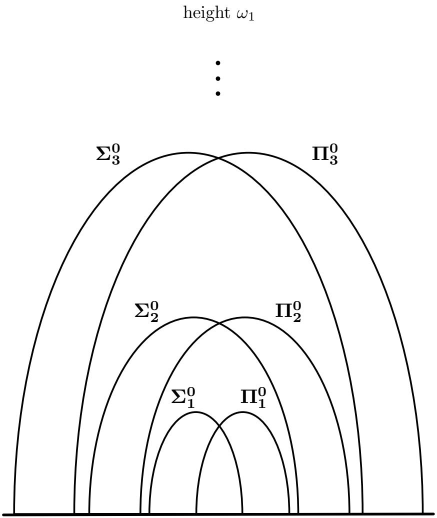
Topology



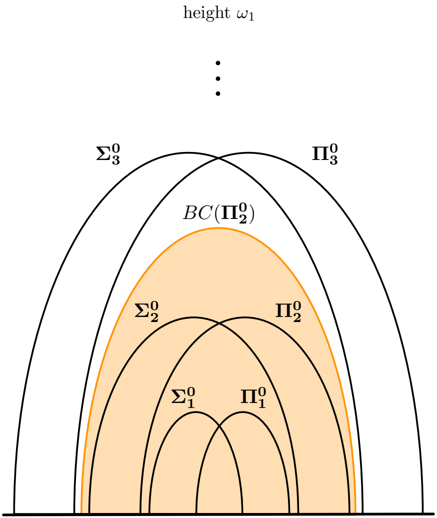
Topology


$\boldsymbol{\Sigma_3^0}$
$\boldsymbol{\Sigma_2^0}$
$\boldsymbol{\Pi_2^0}$
$\boldsymbol{\Sigma_1^0}$
$\boldsymbol{\Pi_1^0}$

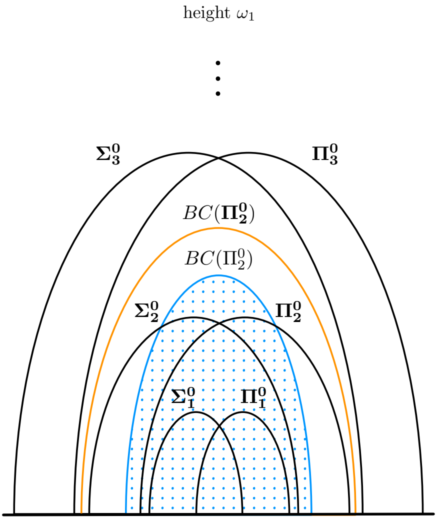
Topology


height ω_1
$\vdots$
$\mathbf{\Sigma^0_3}$
$\mathbf{\Pi^0_3}$
$\mathbf{\Sigma^0_2}$
$\mathbf{\Pi^0_2}$
$\mathbf{\Sigma^0_1}$
$\mathbf{\Pi^0_1}$

Topology


height ω_1
⋮
$\mathbf{\Sigma_3^0}$
$\mathbf{\Pi_3^0}$
$BC(\mathbf{\Pi_2^0})$
$\mathbf{\Sigma_2^0}$
$\mathbf{\Pi_2^0}$
$\mathbf{\Sigma_1^0}$
$\mathbf{\Pi_1^0}$

Topology



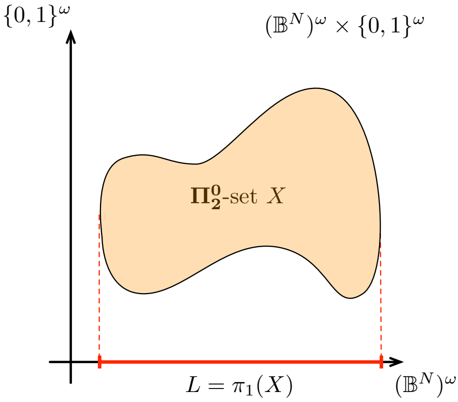
height ω_1
⋮
$\boldsymbol{\Sigma}^0_3$
$\boldsymbol{\Pi}^0_3$
$BC(\boldsymbol{\Pi}^0_2)$
$BC(\Pi^0_2)$
$\boldsymbol{\Sigma}^0_2$
$\boldsymbol{\Pi}^0_2$
$\boldsymbol{\Sigma}^0_1$
$\boldsymbol{\Pi}^0_1$

Topology

$$\begin{tikzcd}[column sep=small]
& \boldsymbol{\Delta}^1_1 & \\
& \vdots & \\
\boldsymbol{\Sigma}^0_3 & & \boldsymbol{\Pi}^0_3 \\
& \boldsymbol{\Delta}^0_3 \arrow[ul, phantom, sloped, "\subsetneq"] \arrow[ur, phantom, sloped, "\subsetneq"] & \\
\boldsymbol{\Sigma}^0_2 \arrow[ur, phantom, sloped, "\subsetneq"] & & \boldsymbol{\Pi}^0_2 \arrow[ul, phantom, sloped, "\subsetneq"] \\
& \boldsymbol{\Delta}^0_2 \arrow[ul, phantom, sloped, "\subsetneq"] \arrow[ur, phantom, sloped, "\subsetneq"] & \\
\boldsymbol{\Sigma}^0_1 \arrow[ur, phantom, sloped, "\subsetneq"] & & \boldsymbol{\Pi}^0_1 \arrow[ul, phantom, sloped, "\subsetneq"] \\
& \boldsymbol{\Delta}^0_1 \arrow[ul, phantom, sloped, "\subsetneq"] \arrow[ur, phantom, sloped, "\subsetneq"] &
\end{tikzcd}$$

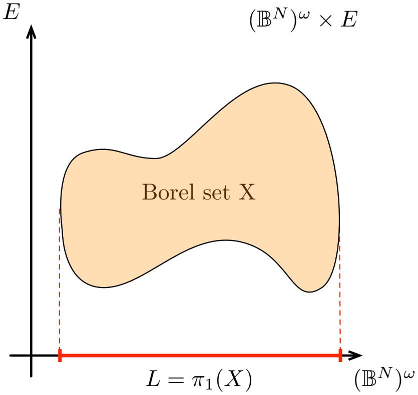
Topology

- An ω-language $L \subseteq (\mathbb{B}^N)^\omega$ is *analytic* ($\boldsymbol{\Sigma}_1^1$) iff it is the first projection of some $\boldsymbol{\Pi}_2^0$-set $X \subseteq (\mathbb{B}^N)^\omega \times \{0,1\}^\omega$.



Topology

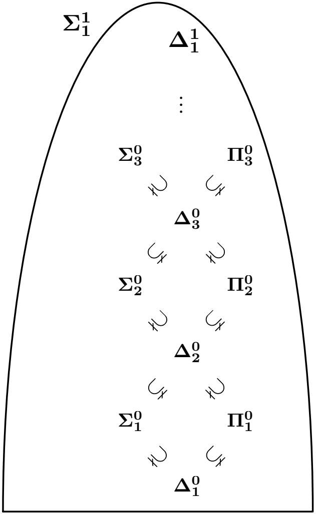
- An ω-language $L \subseteq (\mathbb{B}^N)^\omega$ is *analytic* ($\mathbf{\Sigma}^1_1$) iff it is the first projection of some Borel set $X \subseteq (\mathbb{B}^N)^\omega \times E$, where E is a Polish space.



Topology

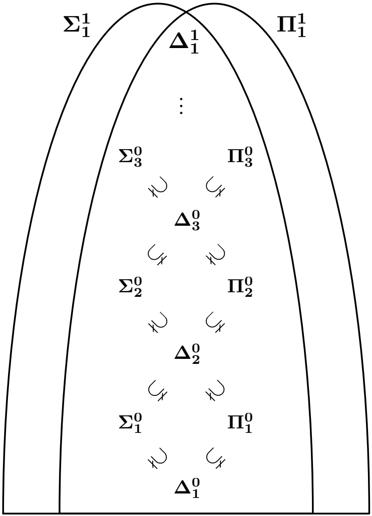
$$\begin{tikzcd}[column sep=small]
& \boldsymbol{\Delta}^1_1 & \\
& \vdots & \\
\boldsymbol{\Sigma}^0_3 & & \boldsymbol{\Pi}^0_3 \\
& \boldsymbol{\Delta}^0_3 \arrow[ul, phantom, sloped, "\subsetneq"] \arrow[ur, phantom, sloped, "\subsetneq"] & \\
\boldsymbol{\Sigma}^0_2 \arrow[ur, phantom, sloped, "\subsetneq"] & & \boldsymbol{\Pi}^0_2 \arrow[ul, phantom, sloped, "\subsetneq"] \\
& \boldsymbol{\Delta}^0_2 \arrow[ul, phantom, sloped, "\subsetneq"] \arrow[ur, phantom, sloped, "\subsetneq"] & \\
\boldsymbol{\Sigma}^0_1 \arrow[ur, phantom, sloped, "\subsetneq"] & & \boldsymbol{\Pi}^0_1 \arrow[ul, phantom, sloped, "\subsetneq"] \\
& \boldsymbol{\Delta}^0_1 \arrow[ul, phantom, sloped, "\subsetneq"] \arrow[ur, phantom, sloped, "\subsetneq"] &
\end{tikzcd}$$

Topology

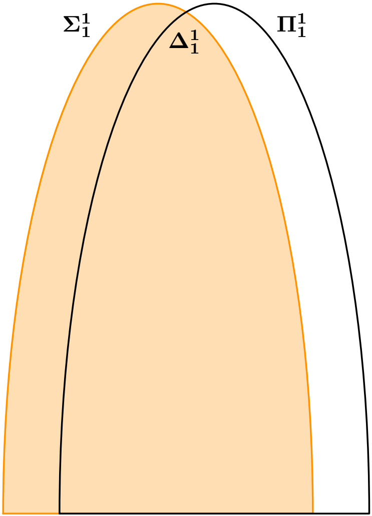


$\boldsymbol{\Sigma}^1_1$
$\boldsymbol{\Delta}^1_1$
$\vdots$
$\boldsymbol{\Sigma}^0_3$ $\boldsymbol{\Pi}^0_3$
$\boldsymbol{\Delta}^0_3$
$\boldsymbol{\Sigma}^0_2$ $\boldsymbol{\Pi}^0_2$
$\boldsymbol{\Delta}^0_2$
$\boldsymbol{\Sigma}^0_1$ $\boldsymbol{\Pi}^0_1$
$\boldsymbol{\Delta}^0_1$

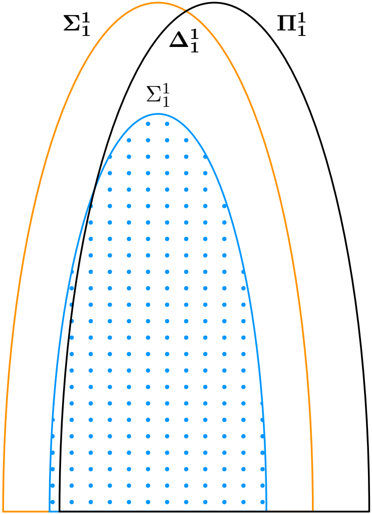
Topology


$\boldsymbol{\Sigma}^1_1$
$\boldsymbol{\Pi}^1_1$
$\boldsymbol{\Delta}^1_1$
$\vdots$
$\boldsymbol{\Sigma}^0_3$
$\boldsymbol{\Pi}^0_3$
$\boldsymbol{\Delta}^0_3$
$\boldsymbol{\Sigma}^0_2$
$\boldsymbol{\Pi}^0_2$
$\boldsymbol{\Delta}^0_2$
$\boldsymbol{\Sigma}^0_1$
$\boldsymbol{\Pi}^0_1$
$\boldsymbol{\Delta}^0_1$

Topology

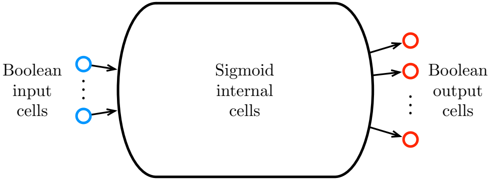


Topology



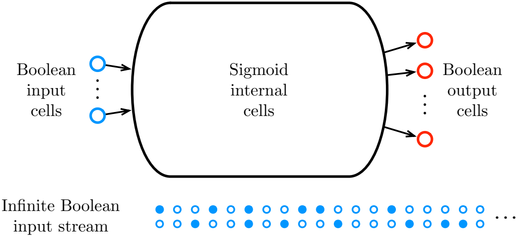
Deterministic ω-RNNs

We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



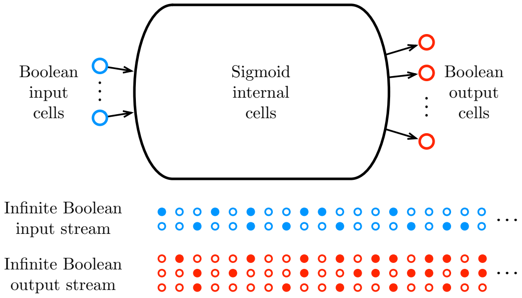
Deterministic ω-RNNs

We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



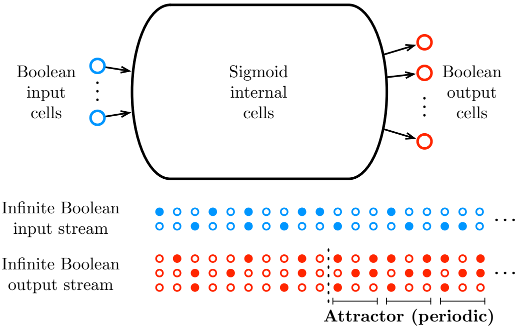
Deterministic ω-RNNs

We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



Deterministic ω-RNNs

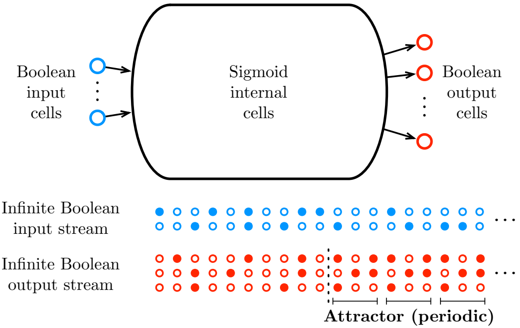
We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



- The attractors are assumed to be classified into two possible kinds: *accepting* or *rejecting*.

Deterministic ω-RNNs

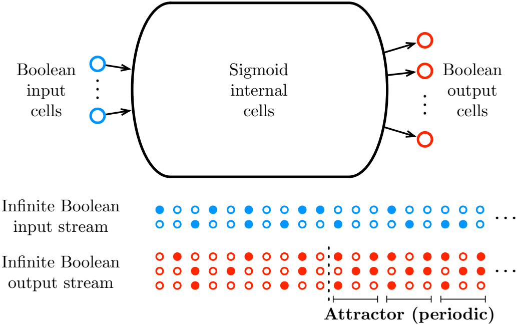
We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



- An infinite Boolean input stream is *accepted* by $\mathcal{N}$ if the corresponding Boolean output stream visits a *accepting* attractor.

Deterministic ω-RNNs

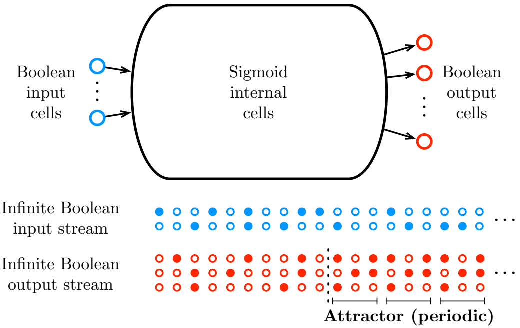
We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



- An infinite Boolean input stream is *rejected* by $\mathcal{N}$ if the corresponding Boolean output stream visits a *rejecting* attractor.

Deterministic ω-RNNs

We consider RNNs with Boolean input and output cells, sigmoidal internal cells, and working on infinite input streams.



- The set of all input streams that are accepted by $\mathcal{N}$ is the *ω-language* recognized by $\mathcal{N}$.

Deterministic ω-RNNs

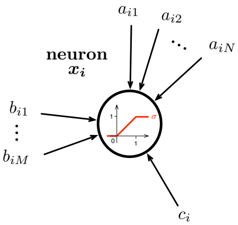
We consider six models of deterministic RNNs:

1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s

Deterministic ω-RNNs

We consider six models of deterministic RNNs:

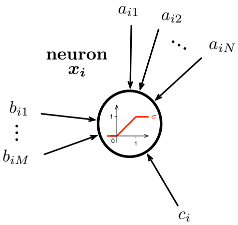
1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s



Deterministic ω-RNNs

We consider six models of deterministic RNNs:

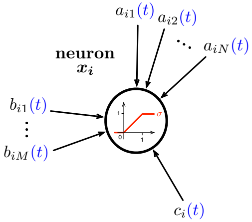
1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s



Deterministic ω-RNNs

We consider six models of deterministic RNNs:

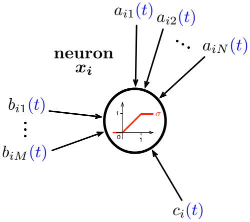
1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s



Deterministic ω-RNNs

We consider six models of deterministic RNNs:

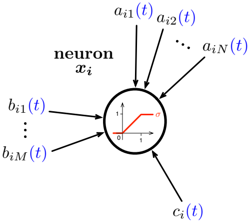
1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s



Deterministic ω-RNNs

We consider six models of deterministic RNNs:

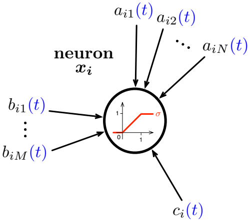
1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s



Deterministic ω-RNNs

We consider six models of deterministic RNNs:

1. static rational RNNs: D-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: D-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: D-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: D-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: D-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: D-Ev-RNN[$\mathbb{R}$]s



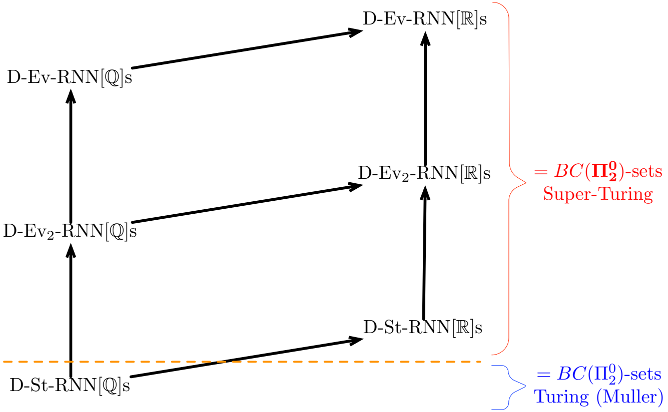
Results

Relationship between the models:

$$\begin{tikzcd}
 & \text{D-Ev-RNN}[\mathbb{R}]\text{s} \\
\text{D-Ev-RNN}[\mathbb{Q}]\text{s} \arrow[ur] & \\
 & \text{D-Ev}_2\text{-RNN}[\mathbb{R}]\text{s} \arrow[uu] \\
\text{D-Ev}_2\text{-RNN}[\mathbb{Q}]\text{s} \arrow[ur] \arrow[uu] & \\
 & \text{D-St-RNN}[\mathbb{R}]\text{s} \arrow[uu] \\
\text{D-St-RNN}[\mathbb{Q}]\text{s} \arrow[ur] \arrow[uu] &
\end{tikzcd}$$

Results

Relationship between the models:



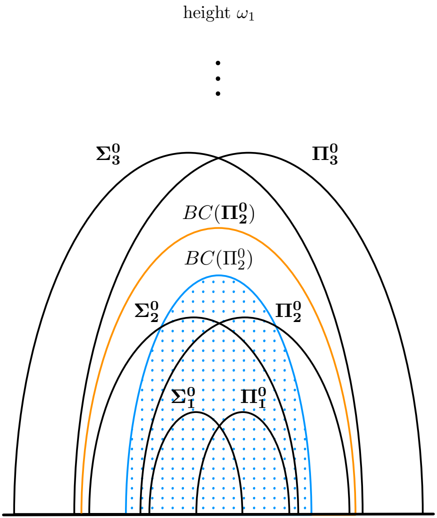
Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in BC(\Pi^0_2)$
- *L is recognizable by some deterministic Muller TM*
- *L is recognizable by some D-St-RNN[$\mathbb{Q}$]*

Results



height ω_1
$\vdots$
$\mathbf{\Sigma^0_3}$
$\mathbf{\Pi^0_3}$
$BC(\mathbf{\Pi^0_2})$
$BC(\Pi^0_2)$
$\mathbf{\Sigma^0_2}$
$\mathbf{\Pi^0_2}$
$\mathbf{\Sigma^0_1}$
$\mathbf{\Pi^0_1}$

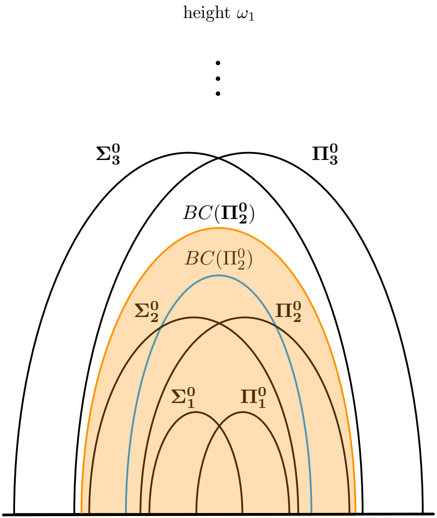
Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in BC(\mathbf{\Pi^0_2})$;
- L *is recognizable by some D-St-RNN[$\mathbb{R}$];*
- L *is recognizable by some D-Ev$_2$-RNN[$\mathbb{Q}$];*
- L *is recognizable by some D-Ev-RNN[$\mathbb{Q}$];*
- L *is recognizable by some D-Ev$_2$-RNN[$\mathbb{R}$];*
- L *is recognizable by some D-Ev-RNN[$\mathbb{R}$].*

Results


height ω_1
$\vdots$
$\mathbf{\Sigma^0_3}$
$\mathbf{\Pi^0_3}$
$BC(\mathbf{\Pi^0_2})$
$BC(\Pi^0_2)$
$\mathbf{\Sigma^0_2}$
$\mathbf{\Pi^0_2}$
$\mathbf{\Sigma^0_1}$
$\mathbf{\Pi^0_1}$

Results – Summary

DET.	Static	Bi-valued Evolving	General Evolving
$\mathbb{Q}$	D-St-RNN[$\mathbb{Q}$]s $= BC(\Pi^0_2)$ Turing (Muller)	D-Ev$_2$-RNN[$\mathbb{Q}$]s $= BC(\mathbf{\Pi^0_2})$ super-Turing	D-Ev-RNN[$\mathbb{Q}$]s $= BC(\mathbf{\Pi^0_2})$ super-Turing
$\mathbb{R}$	D-St-RNN[$\mathbb{R}$]s $= BC(\mathbf{\Pi^0_2})$ super-Turing	D-Ev$_2$-RNN[$\mathbb{R}$]s $= BC(\mathbf{\Pi^0_2})$ super-Turing	D-Ev-RNN[$\mathbb{R}$]s $= BC(\mathbf{\Pi^0_2})$ super-Turing

Deterministic ω-RNNs

We consider two other models of deterministic RNNs:

1. static real RNNs with only one real weight:

   $$\text{D-St-RNN}[\mathbb{Q}, \ulcorner \alpha \urcorner]\text{s, where } \ulcorner \alpha \urcorner \in \mathbb{R} \text{ encoding of } \alpha \in \{0,1\}^\omega$$

2. bi-valued evolving rational RNNs with only one evolving weight:

   $$\text{D-Ev}_2\text{-RNN}[\mathbb{Q}, \alpha]\text{s, where } \alpha \in \{0,1\}^\omega$$

Deterministic ω-RNNs

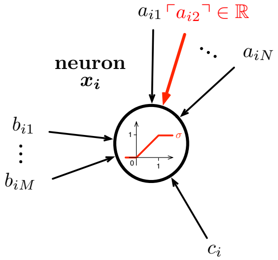
We consider two other models of deterministic RNNs:

1. static real RNNs with only one real weight:

   D-St-RNN[$\mathbb{Q}, \ulcorner\alpha\urcorner$]s, where $\ulcorner\alpha\urcorner \in \mathbb{R}$ encoding of $\alpha \in \{0,1\}^\omega$

2. bi-valued evolving rational RNNs with only one evolving weight:

   D-Ev$_2$-RNN[$\mathbb{Q}, \alpha$]s, where $\alpha \in \{0,1\}^\omega$



Deterministic ω-RNNs

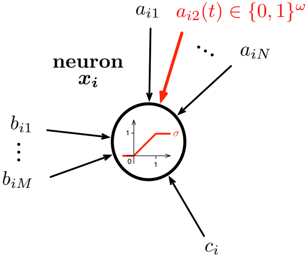
We consider two other models of deterministic RNNs:

1. static real RNNs with only one real weight:

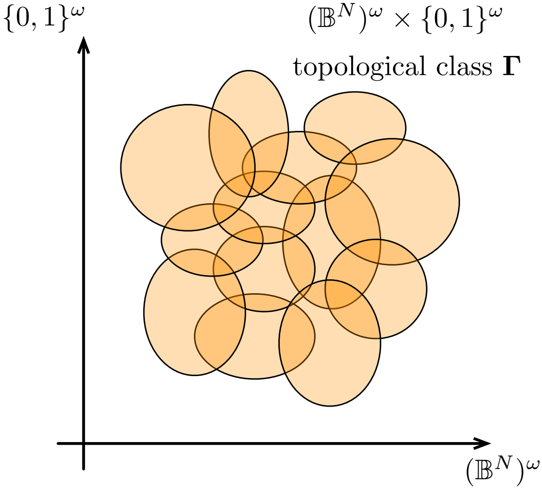
   D-St-RNN$[\mathbb{Q}, \ulcorner\alpha\urcorner]$s, where $\ulcorner\alpha\urcorner \in \mathbb{R}$ encoding of $\alpha \in \{0,1\}^\omega$

2. bi-valued evolving rational RNNs with only one evolving weight:

   D-Ev$_2$-RNN$[\mathbb{Q}, \alpha]$s, where $\alpha \in \{0,1\}^\omega$

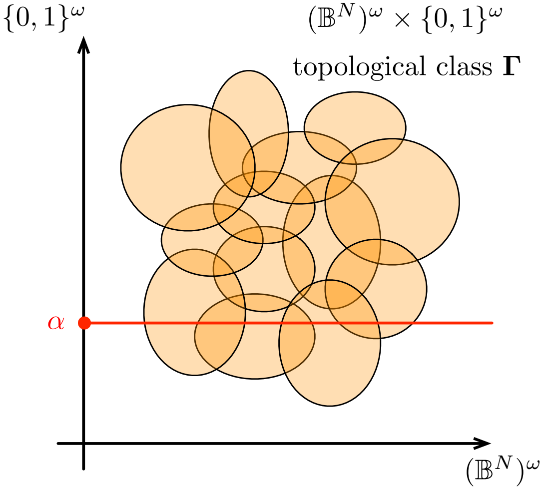


Topology

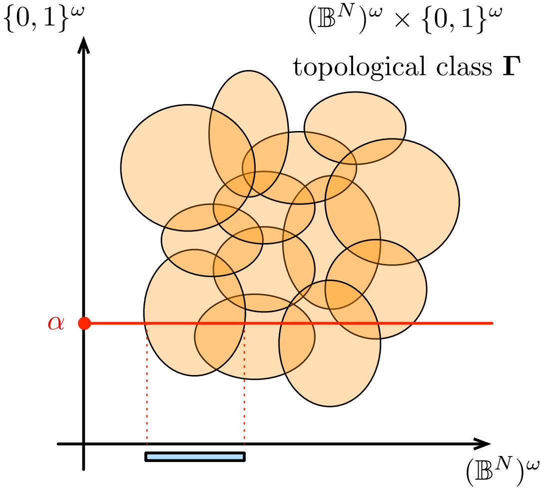


$\{0,1\}^\omega$
$(\mathbb{B}^N)^\omega \times \{0,1\}^\omega$
topological class $\mathbf{\Gamma}$
$(\mathbb{B}^N)^\omega$

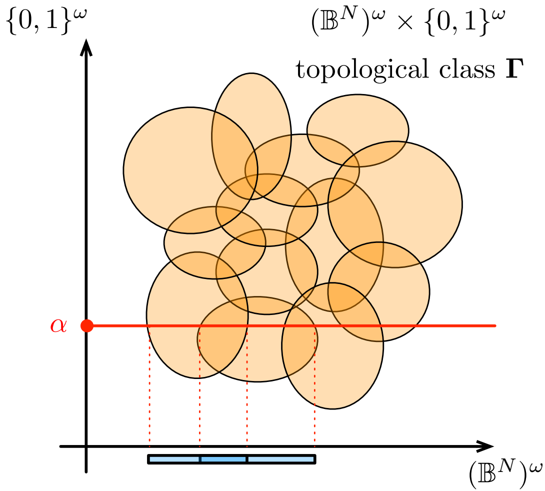
Topology


$\{0,1\}^\omega$
$(\mathbb{B}^N)^\omega \times \{0,1\}^\omega$
topological class $\mathbf{\Gamma}$
α
$(\mathbb{B}^N)^\omega$

Topology

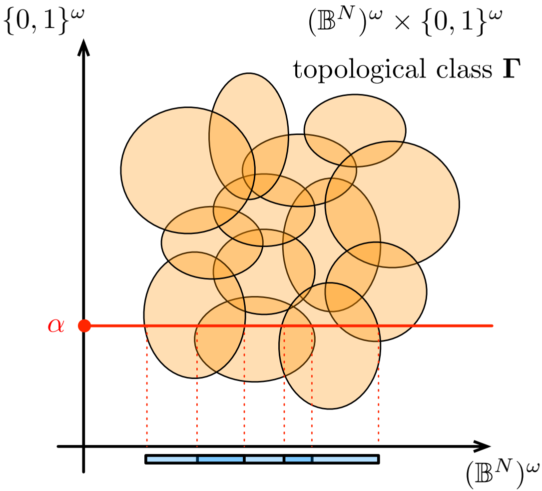

$\{0,1\}^\omega$
$(\mathbb{B}^N)^\omega \times \{0,1\}^\omega$
topological class $\mathbf{\Gamma}$
α
$(\mathbb{B}^N)^\omega$

Topology

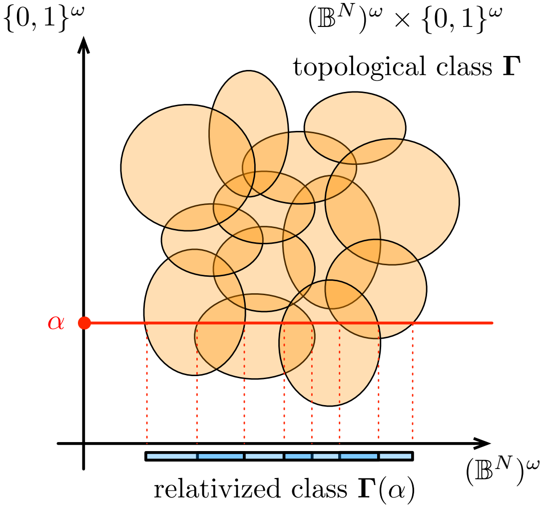


$\{0,1\}^\omega$
$(\mathbb{B}^N)^\omega \times \{0,1\}^\omega$
topological class $\mathbf{\Gamma}$
α
$(\mathbb{B}^N)^\omega$

Topology


$\{0,1\}^\omega$
$(\mathbb{B}^N)^\omega \times \{0,1\}^\omega$
topological class $\mathbf{\Gamma}$
α
$(\mathbb{B}^N)^\omega$

Topology



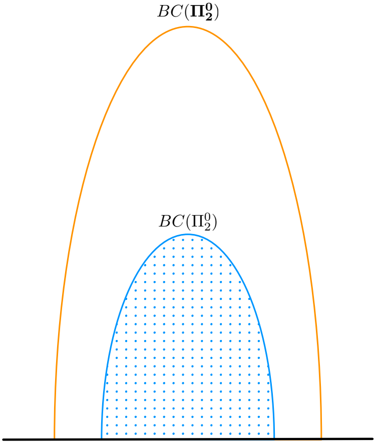
$\{0,1\}^\omega$
$(\mathbb{B}^N)^\omega \times \{0,1\}^\omega$
topological class $\mathbf{\Gamma}$
α
$(\mathbb{B}^N)^\omega$
relativized class $\mathbf{\Gamma}(\alpha)$

Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in BC(\Pi^0_2)(\alpha)$, *for some* $\alpha \in \{0,1\}^\omega$
- L *is recognizable by some D-St-RNN*[$\mathbb{Q}, \ulcorner \alpha \urcorner$]
- L *is recognizable by some D-Ev$_2$-RNN*[$\mathbb{Q}, \alpha$]

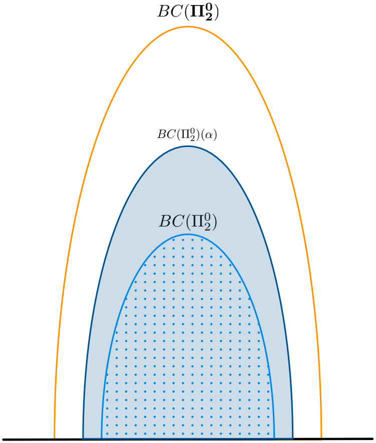


Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in BC(\Pi^0_2)(\alpha)$, *for some* $\alpha \in \{0,1\}^\omega$
- L *is recognizable by some D-St-RNN*[$\mathbb{Q}, \ulcorner\alpha\urcorner$]
- L *is recognizable by some D-Ev*$_2$*-RNN*[$\mathbb{Q}, \alpha$]



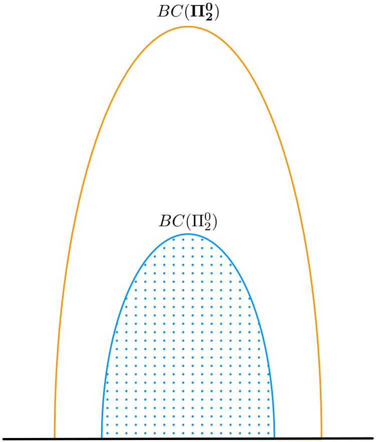
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, where each $\alpha_i \in \{0,1\}^\omega$, such that

$$BC(\Pi^0_2)(\alpha_i) \subsetneq BC(\Pi^0_2)(\alpha_j)$$

for all $i < j < \omega_1$.



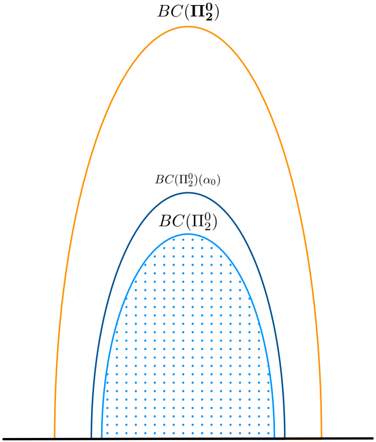
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, where each $\alpha_i \in \{0,1\}^\omega$, such that

$$BC(\Pi^0_2)(\alpha_i) \subsetneq BC(\Pi^0_2)(\alpha_j)$$

for all $i < j < \omega_1$.



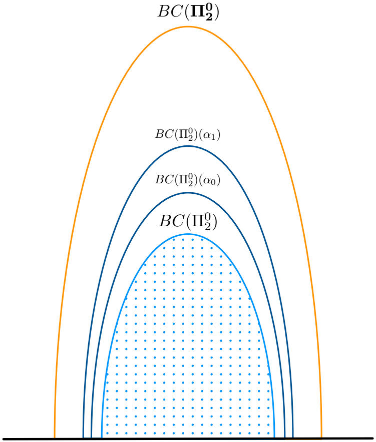
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, *where each* $\alpha_i \in \{0,1\}^\omega$, *such that*

$$BC(\Pi^0_2)(\alpha_i) \subsetneq BC(\Pi^0_2)(\alpha_j)$$

for all $i < j < \omega_1$.



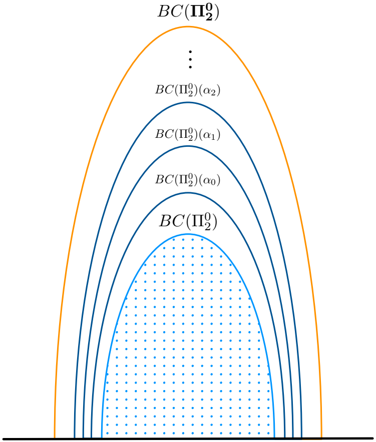
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, where each $\alpha_i \in \{0,1\}^\omega$, such that

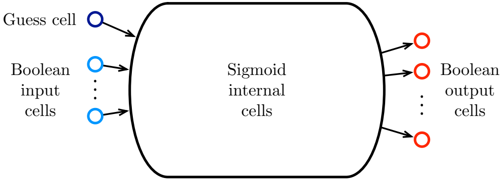
$$BC(\Pi^0_2)(\alpha_i) \subsetneq BC(\Pi^0_2)(\alpha_j)$$

for all $i < j < \omega_1$.



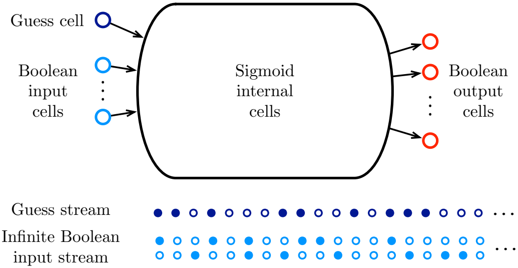
Nondeterministic ω-RNNs (Type I)

The RNNs are provided with an additional Boolean guess cell.



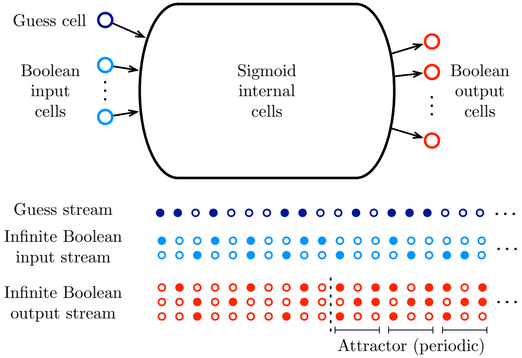
Nondeterministic ω-RNNs (Type I)

The RNNs are provided with an additional Boolean guess cell.



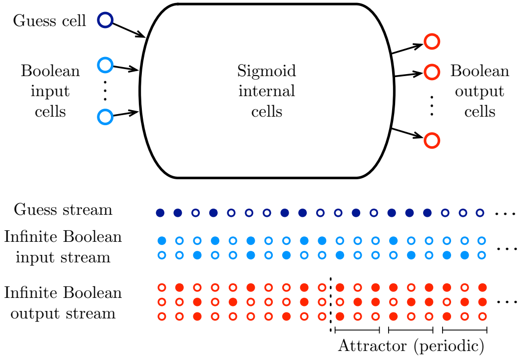
Nondeterministic ω-RNNs (Type I)

The RNNs are provided with an additional Boolean guess cell.



Nondeterministic ω-RNNs (Type I)

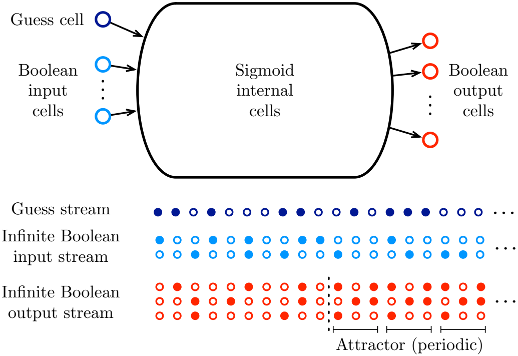
The RNNs are provided with an additional Boolean guess cell.



- The attractors are assumed to be classified into two possible kinds: *accepting* or *rejecting*.

Nondeterministic ω-RNNs (Type I)

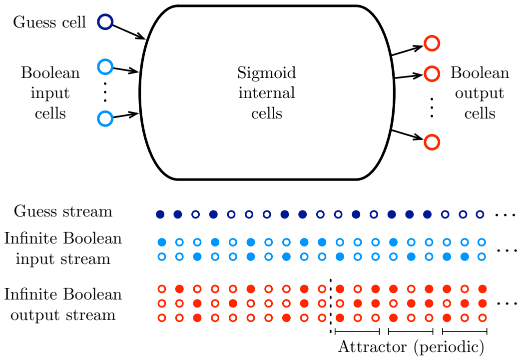
The RNNs are provided with an additional Boolean guess cell.



- Input stream $s \in (\mathbb{B}^M)^\omega$ *accepted* by $\mathcal{N}$ iff there exists some guess $g \in \mathbb{B}^\omega$ s.t. $\mathcal{N}(s, g)$ enters an accepting attractor.

Nondeterministic ω-RNNs (Type I)

The RNNs are provided with an additional Boolean guess cell.



- Input stream $s \in (\mathbb{B}^M)^\omega$ *rejected* by $\mathcal{N}$ iff for all guess $g \in \mathbb{B}^\omega$, $\mathcal{N}(s, g)$ enters an accepting attractor.

Nondeterministic ω-RNNs (Type I)

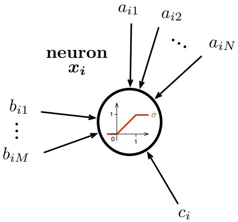
We consider six models of nondeterministic RNNs (of type I):

1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s

Nondeterministic ω-RNNs (Type I)

We consider six models of nondeterministic RNNs (of type I):

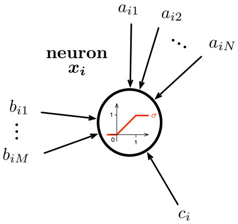
1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s



Nondeterministic ω-RNNs (Type I)

We consider six models of nondeterministic RNNs (of type I):

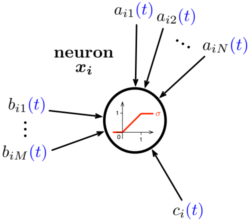
1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s



Nondeterministic ω-RNNs (Type I)

We consider six models of nondeterministic RNNs (of type I):

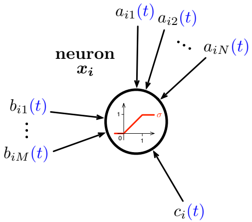
1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s



Nondeterministic ω-RNNs (Type I)

We consider six models of nondeterministic RNNs (of type I):

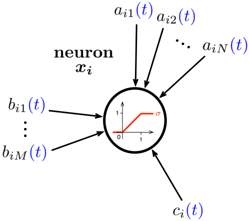
1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s



Nondeterministic ω-RNNs (Type I)

We consider six models of nondeterministic RNNs (of type I):

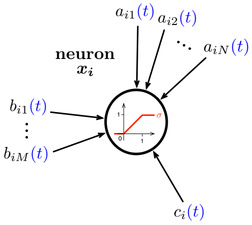
1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s



Nondeterministic ω-RNNs (Type I)

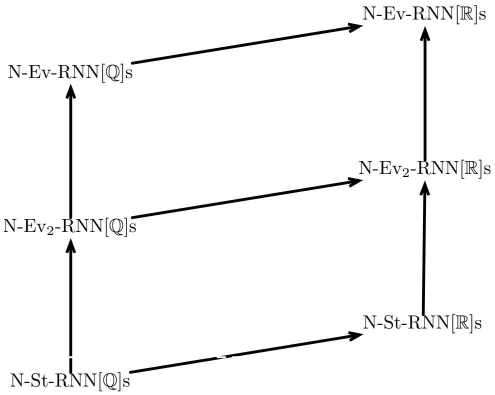
We consider six models of nondeterministic RNNs (of type I):

1. static rational RNNs: N-St-RNN[$\mathbb{Q}$]s
2. static real RNNs: N-St-RNN[$\mathbb{R}$]s
3. bi-valued evolving rational RNNs: N-Ev$_2$-RNN[$\mathbb{Q}$]s
4. bi-valued evolving real RNNs: N-Ev$_2$-RNN[$\mathbb{R}$]s
5. general evolving rational RNNs: N-Ev-RNN[$\mathbb{Q}$]s
6. general evolving real N-RNNs: N-Ev-RNN[$\mathbb{R}$]s



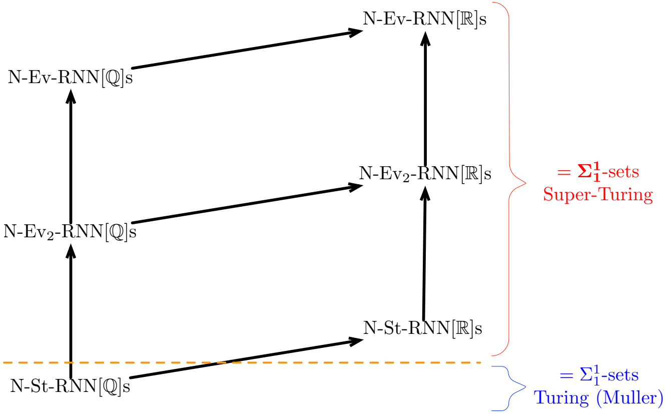
Results

Relationship between the models:



Results

Relationship between the models:



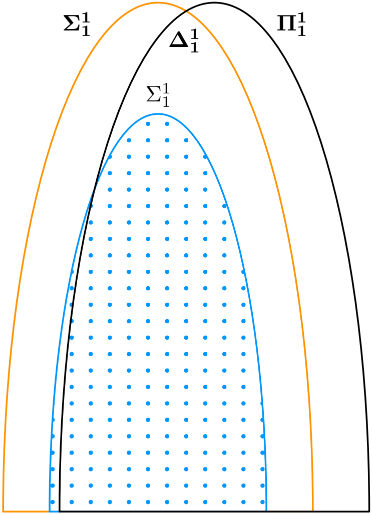
Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in \Sigma^1_1$
- L *is recognizable by some nondeterministic Muller TM*
- L *is recognizable by some N-St-RNN[$\mathbb{Q}$]*

Results



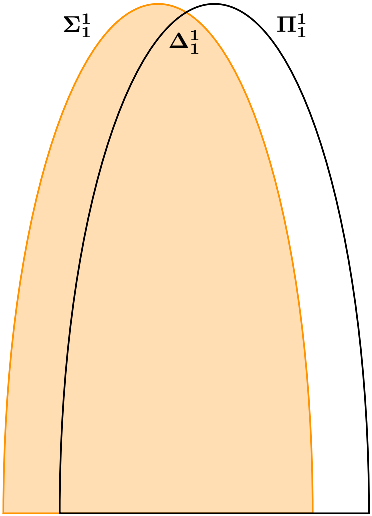
Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in \Sigma^1_1$;
- L *is recognizable by some N-St-RNN[$\mathbb{R}$];*
- L *is recognizable by some N-Ev$_2$-RNN[$\mathbb{Q}$];*
- L *is recognizable by some N-Ev-RNN[$\mathbb{Q}$];*
- L *is recognizable by some N-Ev$_2$-RNN[$\mathbb{R}$];*
- L *is recognizable by some N-Ev-RNN[$\mathbb{R}$].*

Results



Results – Summary

NONDET.	Static	Bi-valued Evolving	General Evolving
$\mathbb{Q}$	N-St-RNN[$\mathbb{Q}$]s $= \Sigma^1_1$ Turing (Muller)	N-Ev$_2$-RNN[$\mathbb{Q}$]s $= \boldsymbol{\Sigma}^1_1$ super-Turing	N-Ev-RNN[$\mathbb{Q}$]s $= \boldsymbol{\Sigma}^1_1$ super-Turing
$\mathbb{R}$	N-St-RNN[$\mathbb{R}$]s $= \boldsymbol{\Sigma}^1_1$ super-Turing	N-Ev$_2$-RNN[$\mathbb{R}$]s $= \boldsymbol{\Sigma}^1_1$ super-Turing	N-Ev-RNN[$\mathbb{R}$]s $= \boldsymbol{\Sigma}^1_1$ super-Turing

Nondeterministic ω-RNNs (Type I)

We consider two other models of nondeterministic RNNs (type I):

1. static real RNNs with only one real weight:

   N-St-RNN[$\mathbb{Q}, \ulcorner\alpha\urcorner$]s, where $\ulcorner\alpha\urcorner \in \mathbb{R}$ encoding of $\alpha \in \{0,1\}^\omega$

2. bi-valued evolving rational RNNs with only one evolving weight:

   N-Ev$_2$-RNN[$\mathbb{Q}, \alpha$]s, where $\alpha \in \{0,1\}^\omega$

Nondeterministic ω-RNNs (Type I)

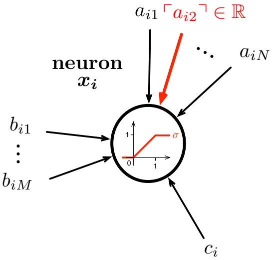
We consider two other models of nondeterministic RNNs (type I):

1. static real RNNs with only one real weight:

   N-St-RNN[$\mathbb{Q}, \ulcorner\alpha\urcorner$]s, where $\ulcorner\alpha\urcorner \in \mathbb{R}$ encoding of $\alpha \in \{0,1\}^\omega$

2. bi-valued evolving rational RNNs with only one evolving weight:

   N-Ev$_2$-RNN[$\mathbb{Q}, \alpha$]s, where $\alpha \in \{0,1\}^\omega$



Nondeterministic ω-RNNs (Type I)

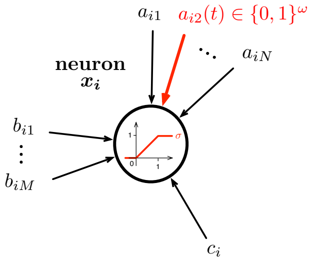
We consider two other models of nondeterministic RNNs (type I):

1. static real RNNs with only one real weight:

   N-St-RNN[$\mathbb{Q}, \ulcorner\alpha\urcorner$]s, where $\ulcorner\alpha\urcorner \in \mathbb{R}$ encoding of $\alpha \in \{0,1\}^\omega$

2. bi-valued evolving rational RNNs with only one evolving weight:

   N-Ev$_2$-RNN[$\mathbb{Q}, \alpha$]s, where $\alpha \in \{0,1\}^\omega$

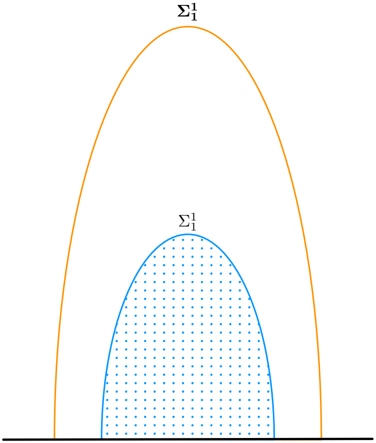


Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in \Sigma^1_1(\alpha)$, *for some* $\alpha \in \{0,1\}^\omega$
- L *is recognizable by some N-St-RNN[$\mathbb{Q}, \ulcorner\alpha\urcorner$]*
- L *is recognizable by some N-Ev$_2$-RNN[$\mathbb{Q}, \alpha$]*

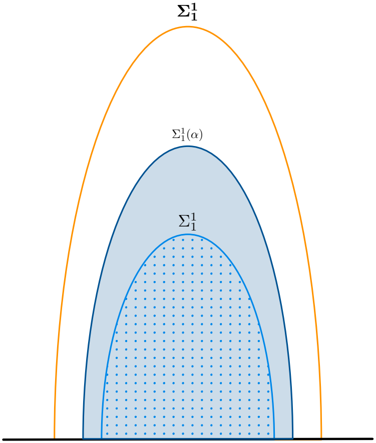


Results

Theorem

Let $L \subseteq (\mathbb{B}^M)^\omega$. The following conditions are equivalent.

- $L \in \Sigma^1_1(\alpha)$, *for some* $\alpha \in \{0,1\}^\omega$
- L *is recognizable by some N-St-RNN[$\mathbb{Q}, \ulcorner\alpha\urcorner$]*
- L *is recognizable by some N-Ev$_2$-RNN[$\mathbb{Q}, \alpha$]*



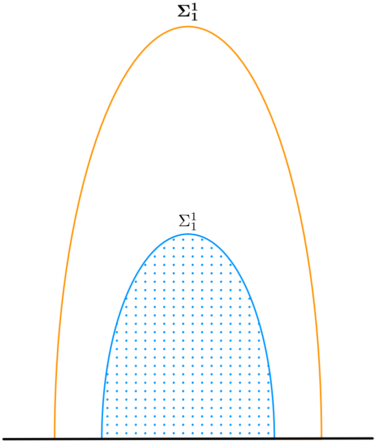
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, where each $\alpha_i \in \{0,1\}^\omega$, such that

$$\Sigma^1_1(\alpha_i) \subsetneq \Sigma^1_1(\alpha_j)$$

for all $i < j < \omega_1$.



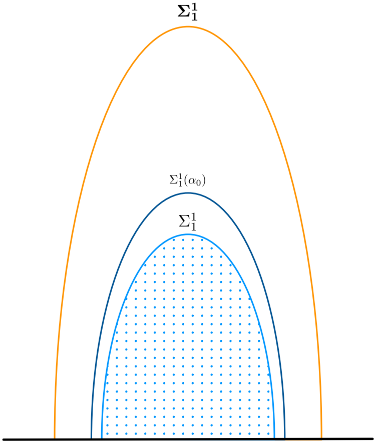
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, where each $\alpha_i \in \{0,1\}^\omega$, such that

$$\Sigma^1_1(\alpha_i) \subsetneq \Sigma^1_1(\alpha_j)$$

for all $i < j < \omega_1$.



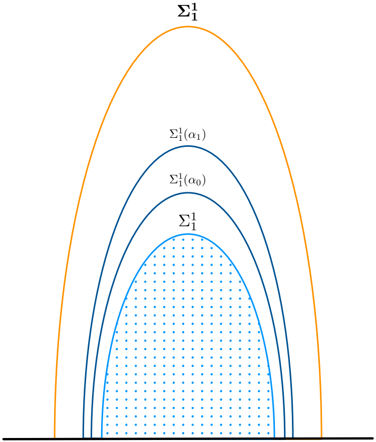
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, *where each* $\alpha_i \in \{0,1\}^\omega$, *such that*

$$\Sigma^1_1(\alpha_i) \subsetneq \Sigma^1_1(\alpha_j)$$

for all $i < j < \omega_1$.



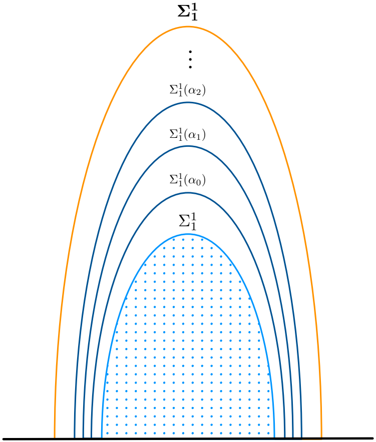
Results

Proposition

There exist some infinite sequence $(\alpha_k)_{k<\omega_1}$, where each $\alpha_i \in \{0,1\}^\omega$, such that

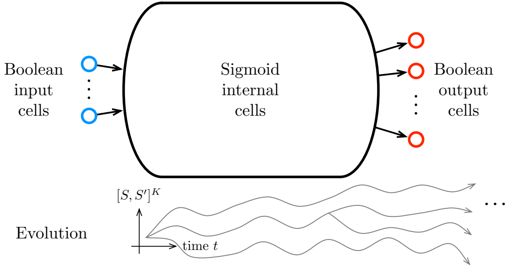
$$\Sigma^1_1(\alpha_i) \subsetneq \Sigma^1_1(\alpha_j)$$

for all $i < j < \omega_1$.



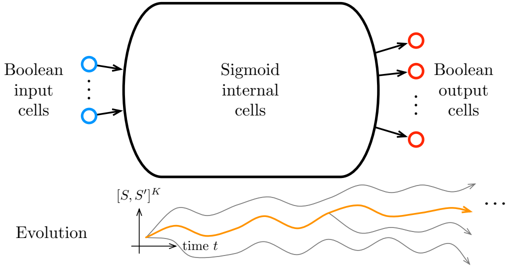
Nondeterministic ω-RNNs (Type II)

The RNNs are provided with an additional evolution set.



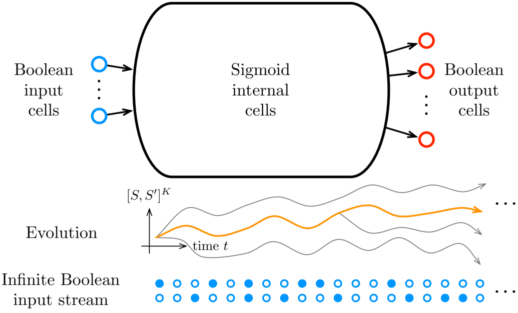
Nondeterministic ω-RNNs (Type II)

The RNNs are provided with an additional evolution set.



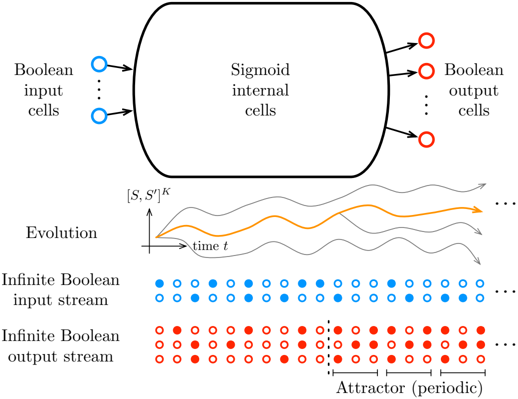
Nondeterministic ω-RNNs (Type II)

The RNNs are provided with an additional evolution set.



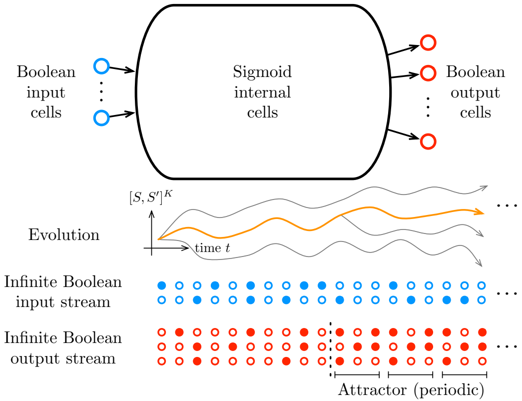
Nondeterministic ω-RNNs (Type II)

The RNNs are provided with an additional evolution set.



Nondeterministic ω-RNNs (Type II)

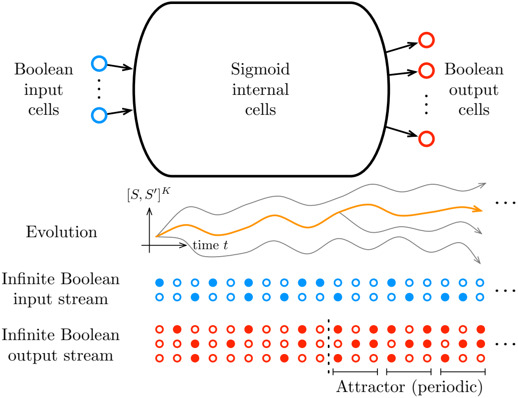
The RNNs are provided with an additional evolution set.



- The attractors are assumed to be classified into two possible kinds: *accepting* or *rejecting*.

Nondeterministic ω-RNNs (Type II)

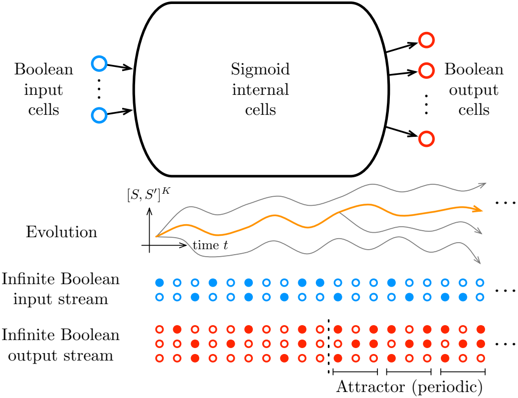
The RNNs are provided with an additional evolution set.



- Input stream $s \in (\mathbb{B}^M)^\omega$ *accepted* by $\mathcal{N}$ iff there exists some evolution $e \in E$ s.t. $\mathcal{N}(s, e)$ enters an accepting attractor.

Nondeterministic ω-RNNs (Type II)

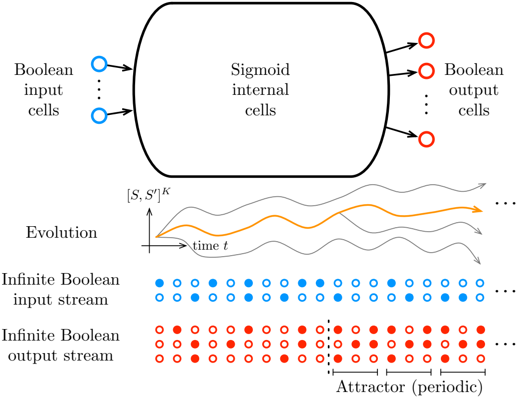
The RNNs are provided with an additional evolution set.



- Input stream $s \in (\mathbb{B}^M)^\omega$ *rejected* by $\mathcal{N}$ iff for all evolutions $e \in E$, $\mathcal{N}(s, e)$ enters an rejecting attractor.

Nondeterministic ω-RNNs (Type II)

The RNNs are provided with an additional evolution set.



- We have more results in this context also...

Conclusion

- We provided a characterization of the expressive power of recurrent neural networks in terms of their attractor dynamics.
- The super-Turing computational capabilities of neural models is related to the issue of *hypercomputation*.
- Current physical theories are consistent with the possibility of hypercomputational systems (quantum, relativistic, etc.). No such systems are currently feasible or harnessable.

Conclusion

- We provided a characterization of the expressive power of recurrent neural networks in terms of their attractor dynamics.
- The super-Turing computational capabilities of neural models is related to the issue of *hypercomputation*.
- Current physical theories are consistent with the possibility of hypercomputational systems (quantum, relativistic, etc.). No such systems are currently feasible or harnessable.

Conclusion

- We provided a characterization of the expressive power of recurrent neural networks in terms of their attractor dynamics.
- The super-Turing computational capabilities of neural models is related to the issue of *hypercomputation*.
- Current physical theories are consistent with the possibility of hypercomputational systems (quantum, relativistic, etc.). No such systems are currently feasible or harnessable.