An Attractor-Based Complexity Measurement for Boolean Recurrent Neural Networks

Application to a Simplified Model of the Basal Ganglia-Thalamocortical Network

Jérémie Cabessa
Joint work with Alessandro E.P. Villa

Department of Mathematical Economics
University of Paris 2
France

6 October 2014

Introduction

- We introduce an new attractor-based complexity measurement for Boolean recurrent neural networks.
- The measurement reflects the complexity of the attractors' structure of the networks.
- We provide an application of this measurement to a simplified Boolean model of the basal ganglia-thalamocortical network

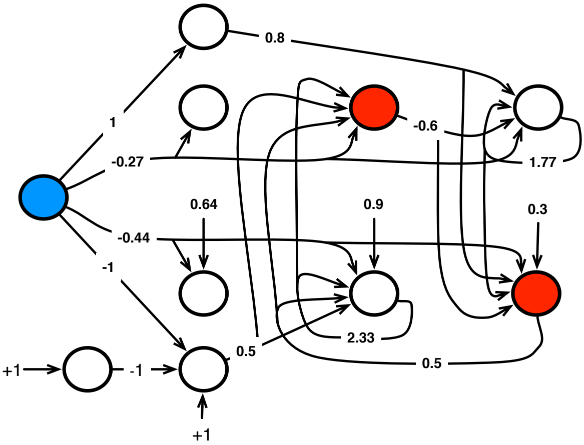
Introduction

- We introduce an new attractor-based complexity measurement for Boolean recurrent neural networks.
- The measurement reflects the complexity of the attractors' structure of the networks.
- We provide an application of this measurement to a simplified Boolean model of the basal ganglia-thalamocortical network

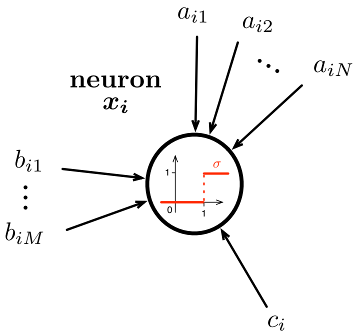
Introduction

- We introduce an new attractor-based complexity measurement for Boolean recurrent neural networks.
- The measurement reflects the complexity of the attractors' structure of the networks.
- We provide an application of this measurement to a simplified Boolean model of the basal ganglia-thalamocortical network

Boolean Recurrent Neural Networks



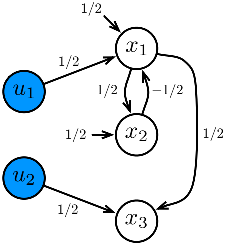
Dynamics



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

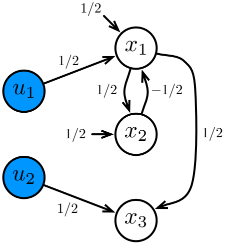
From Boolean Neural Networks to Automata

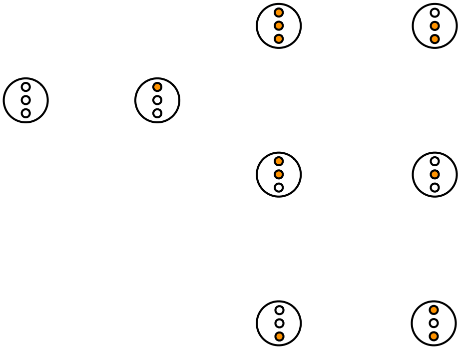
Boolean Neural Network **Automaton**



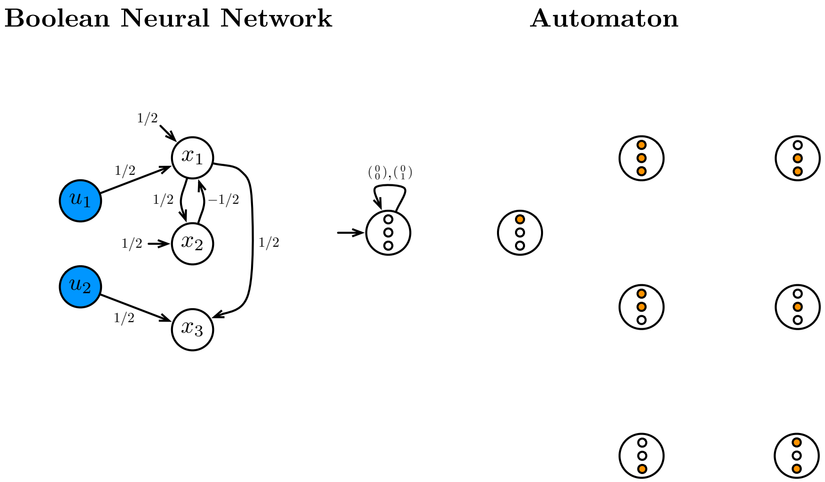
From Boolean Neural Networks to Automata

Boolean Neural Network **Automaton**



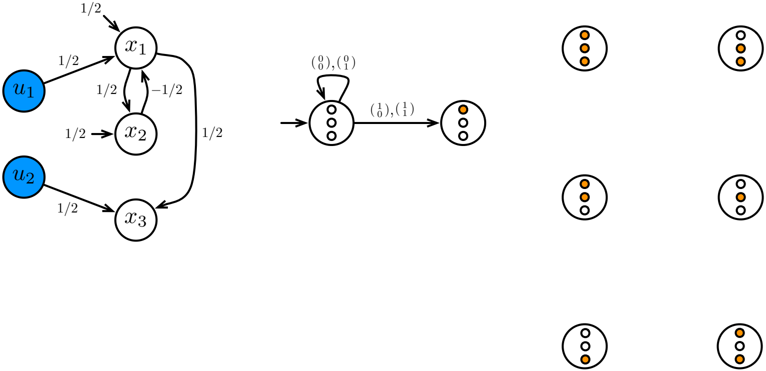


From Boolean Neural Networks to Automata

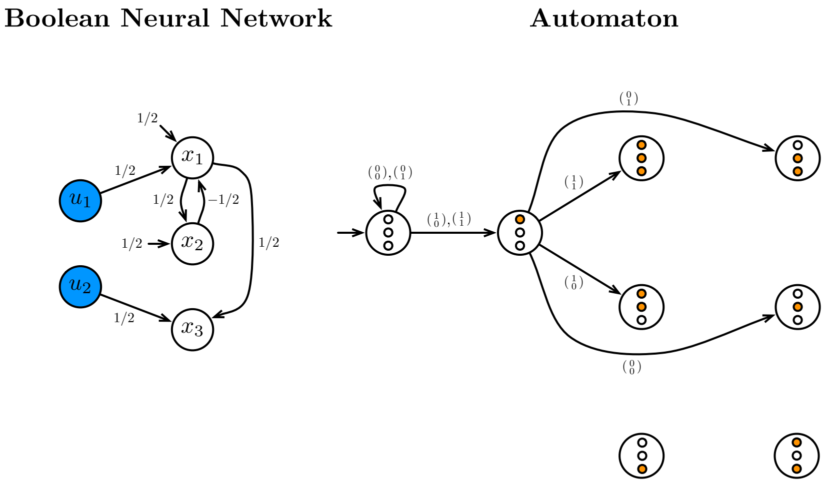


From Boolean Neural Networks to Automata

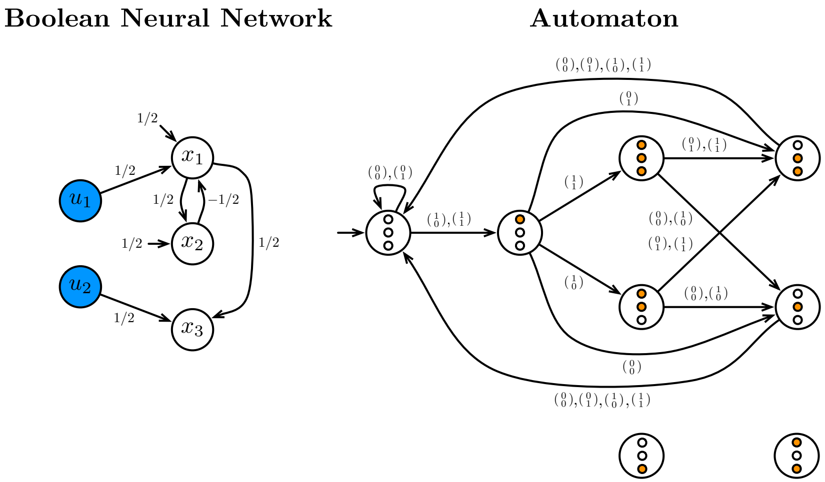
Boolean Neural Network **Automaton**



From Boolean Neural Networks to Automata



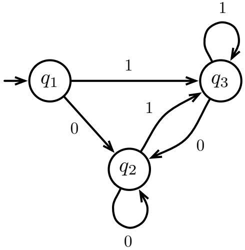
From Boolean Neural Networks to Automata



From Automata to Boolean Neural Networks

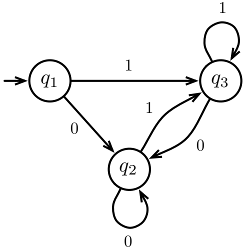
Automaton

Boolean Neural Network

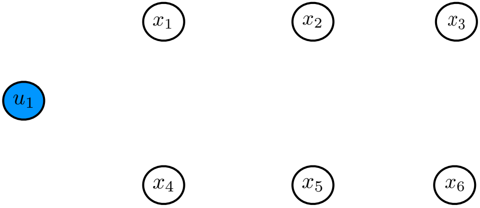


From Automata to Boolean Neural Networks

Automaton

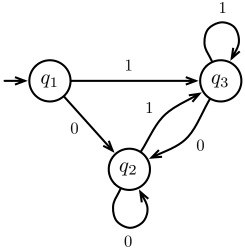


Boolean Neural Network

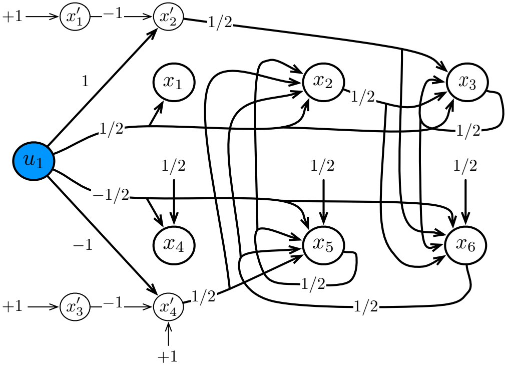


From Automata to Boolean Neural Networks

Automaton



Boolean Neural Network



Equivalence between Boolean Neural Networks and Automata

Theorem (Minsky 67)

"It is evident that each neural network of the kind we have been considering is a finite-state machine."

"[...] It is interesting and even surprising that there is a converse to this. Every finite-state machine is equivalent to, and can be "simulated" by, some neural net."

Equivalence between Boolean Neural Networks and Automata

Theorem (Minsky 67)

"It is evident that each neural network of the kind we have been considering is a finite-state machine."

"[...] It is interesting and even surprising that there is a converse to this. Every finite-state machine is equivalent to, and can be "simulated" by, some neural net."

Equivalence between Boolean Neural Networks and Automata

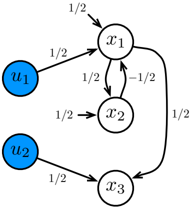
Theorem (Minsky 67)

"It is evident that each neural network of the kind we have been considering is a finite-state machine."

"[...] It is interesting and even surprising that there is a converse to this. Every finite-state machine is equivalent to, and can be "simulated" by, some neural net."

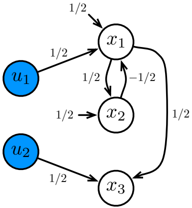
Cycles and Attractors

Boolean Neural Network **Automaton**

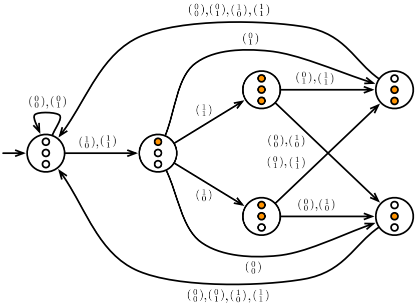


Cycles and Attractors

Boolean Neural Network

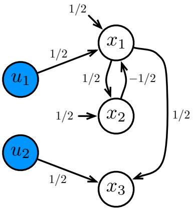


Automaton

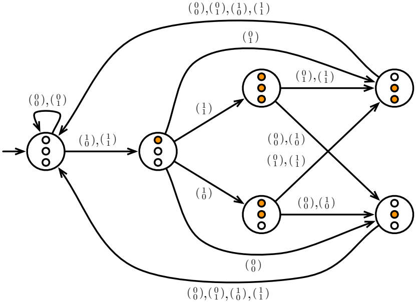


Cycles and Attractors

Boolean Neural Network



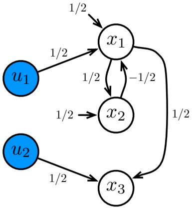
Automaton

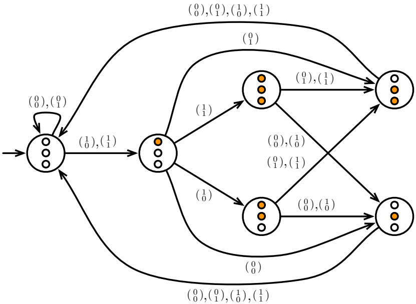


Input stream $\binom{1}{1}\binom{0}{1}\binom{1}{1}\binom{0}{0}\binom{1}{0}\binom{0}{1}\binom{0}{0}\binom{0}{1}\ \cdots$

Cycles and Attractors

Boolean Neural Network | **Automaton**



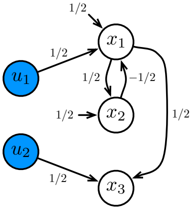


Input stream $\binom{1}{1}\binom{0}{1}\binom{1}{1}\binom{0}{0}\binom{1}{0}\binom{0}{1}\binom{0}{0}\binom{0}{1}\cdots$

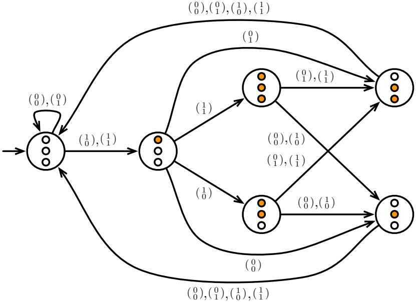
Sequence of states

Cycles and Attractors

Boolean Neural Network

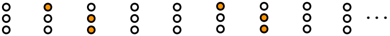


Automaton

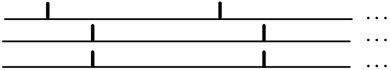


Input stream $\begin{pmatrix}1\\1\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix}\begin{pmatrix}1\\1\end{pmatrix}\begin{pmatrix}0\\0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix}\begin{pmatrix}0\\0\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix}\cdots$

Sequence of states



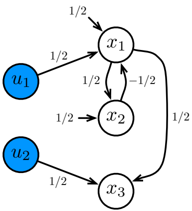
Spatio-temporal pattern

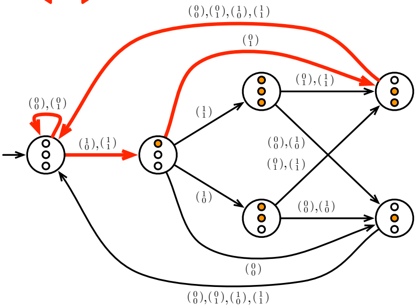


Cycles and Attractors

Boolean Neural Network — **Automaton**

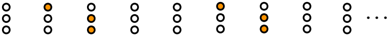
ATTRACTOR ⟺ **CYCLE**



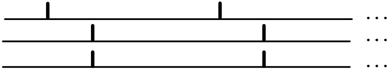


Input stream $\binom{1}{1}\binom{0}{1}\binom{1}{1}\binom{0}{0}\binom{1}{0}\binom{0}{1}\binom{0}{0}\binom{0}{1}\cdots$

Sequence of states



Spatio-temporal pattern



The Wagner Hierarchy

- In ω-automata theory, there is a transfinite classification of ω-automata according to the way their cycles are intricated one into the other...
- **The Wagner hierarchy**
- By translating the Wagner hierarchy from the ω-automata to the Boolean neural network context, one obtains a transfinite classification of Boolean neural networks according to the way their attractors are intricated one into the other...
- **The Boolean RNN hierarchy**

The Wagner Hierarchy

- In ω-automata theory, there is a transfinite classification of ω-automata according to the way their cycles are intricated one into the other...
- **The Wagner hierarchy**
- By translating the Wagner hierarchy from the ω-automata to the Boolean neural network context, one obtains a transfinite classification of Boolean neural networks according to the way their attractors are intricated one into the other...
- **The Boolean RNN hierarchy**

The Wagner Hierarchy

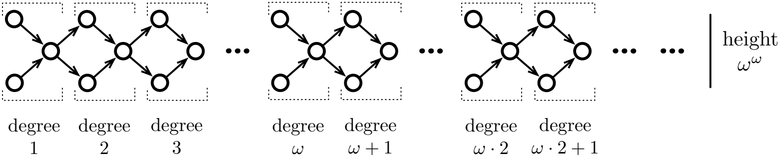
- In ω-automata theory, there is a transfinite classification of ω-automata according to the way their cycles are intricated one into the other...
- **The Wagner hierarchy**
- By translating the Wagner hierarchy from the ω-automata to the Boolean neural network context, one obtains a transfinite classification of Boolean neural networks according to the way their attractors are intricated one into the other...
- **The Boolean RNN hierarchy**

The Wagner Hierarchy

- In ω-automata theory, there is a transfinite classification of ω-automata according to the way their cycles are intricated one into the other...
- **The Wagner hierarchy**
- By translating the Wagner hierarchy from the ω-automata to the Boolean neural network context, one obtains a transfinite classification of Boolean neural networks according to the way their attractors are intricated one into the other...
- **The Boolean RNN hierarchy**

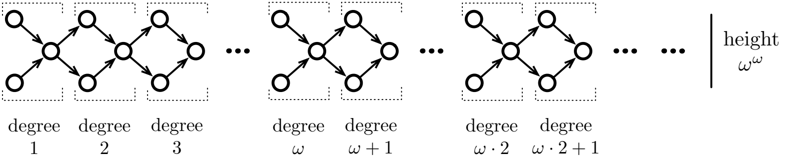
The Wagner Hierarchy

- A transfinite classification of *Muller automata* according to the topological complexity of their underlying language
- Equivalently, a transfinite classification of *Muller automata* according to the graph-theoretical complexity of their cycles
- Quasi well-ordering of transfinite height ω^ω



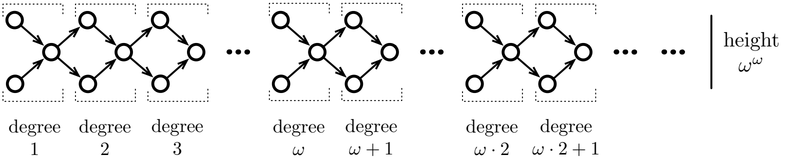
The Wagner Hierarchy

- A transfinite classification of *Muller automata* according to the topological complexity of their underlying language
- Equivalently, a transfinite classification of *Muller automata* according to the graph-theoretical complexity of their cycles
- Quasi well-ordering of transfinite height ω^ω



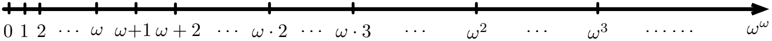
The Wagner Hierarchy

- A transfinite classification of *Muller automata* according to the topological complexity of their underlying language
- Equivalently, a transfinite classification of *Muller automata* according to the graph-theoretical complexity of their cycles
- Quasi well-ordering of transfinite height ω^ω



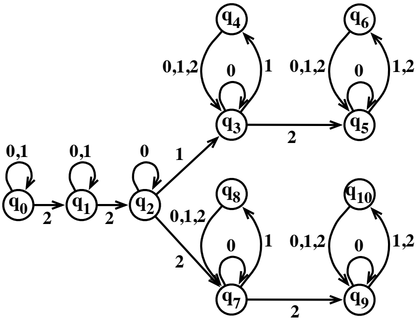
The Wagner Hierarchy

- A transfinite classification of *Muller automata* according to the topological complexity of their underlying language
- Equivalently, a transfinite classification of *Muller automata* according to the graph-theoretical complexity of their cycles
- Quasi well-ordering of transfinite height ω^ω



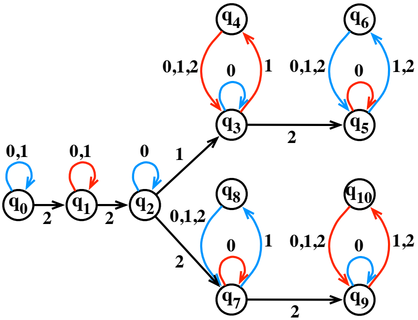
The Wagner Hierarchy – Muller Automata

- A *Muller automaton* consists of an automaton provided with an additional specification of every of its *cycles* into an accepting or a rejecting mode



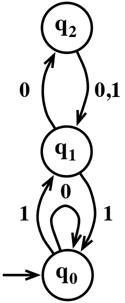
The Wagner Hierarchy – Muller Automata

- A *Muller automaton* consists of an automaton provided with an additional specification of every of its *cycles* into an accepting or a rejecting mode



The Wagner Hierarchy – Degrees ω^n

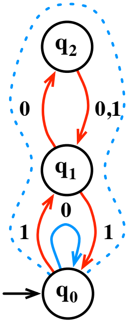
Muller automaton	Wagner degree



The Wagner Hierarchy – Degrees ω^n

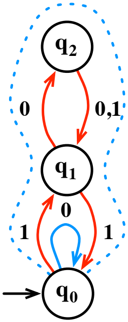
Muller automaton

Wagner degree

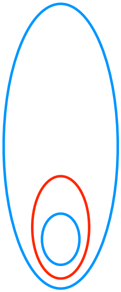


The Wagner Hierarchy – Degrees ω^n

Muller automaton



Wagner degree

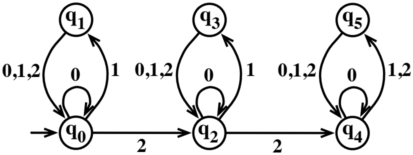


$[+]\ \omega^2$

The Wagner Hierarchy – Degrees $\omega^n \cdot k$

Muller automaton

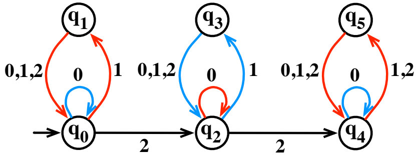
Wagner degree



The Wagner Hierarchy – Degrees $\omega^n \cdot k$

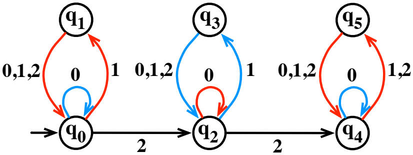
Muller automaton

Wagner degree

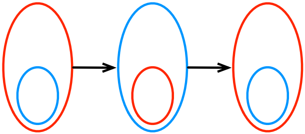


The Wagner Hierarchy – Degrees $\omega^n \cdot k$

Muller automaton



Wagner degree

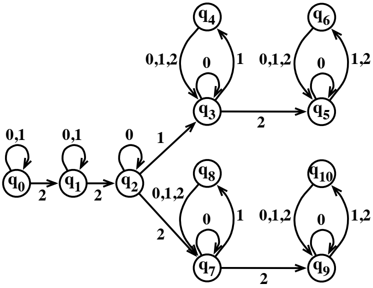


$[+]\ \omega^1 \cdot 3$

The Wagner Hierarchy – Degrees $\omega^n \cdot k + \omega^{n'} \cdot k'$

Muller automaton

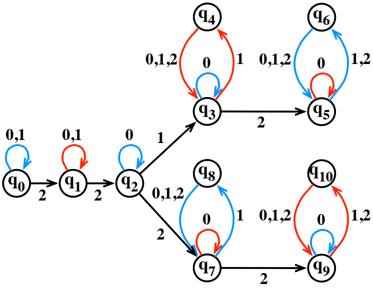
Wagner degree



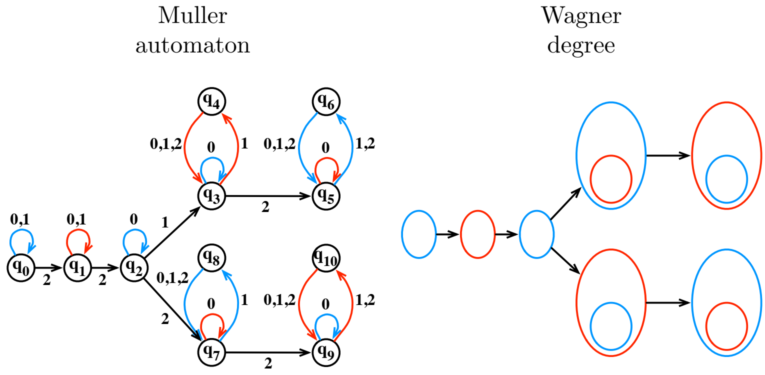
The Wagner Hierarchy – Degrees $\omega^n \cdot k + \omega^{n'} \cdot k'$

Muller automaton

Wagner degree



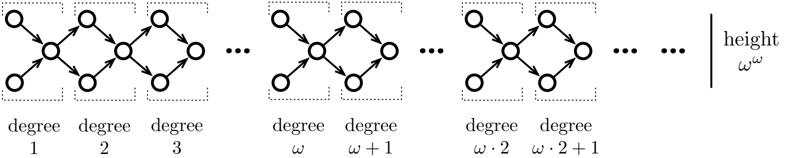
The Wagner Hierarchy – Degrees $\omega^n \cdot k + \omega^{n'} \cdot k'$



$[+]\ \omega^1 \cdot 2 + \omega^0 \cdot 3$

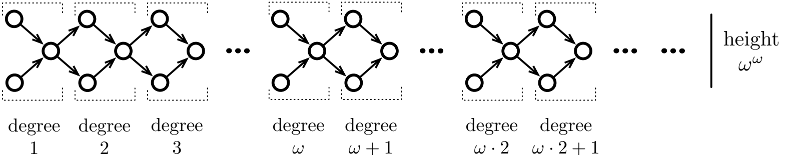
The Wagner Hierarchy – Summary

- A quasi well-ordering of transfinite height ω^ω
- Every ordinal $\alpha < \omega^\omega$ has a unique Cantor normal form $\alpha = \omega^{n_0} \cdot p_0 + \omega^{n_1} \cdot p_1 + \cdots + \omega^{n_k} \cdot p_k$, where $n_0 > n_1 > \cdots > n_k$
- The degree α of a Muller automaton $\mathcal{M}$ in the Wagner hierarchy is the maximal "tree of cycles" $\mathcal{T}_\alpha$ in $\mathcal{M}$



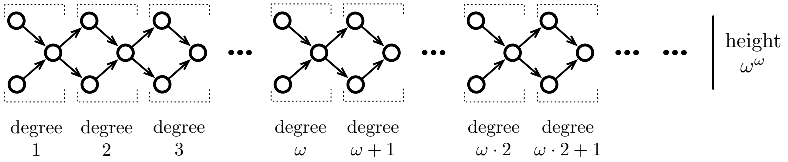
The Wagner Hierarchy – Summary

- A quasi well-ordering of transfinite height ω^ω
- Every ordinal $\alpha < \omega^\omega$ has a unique Cantor normal form $\alpha = \omega^{n_0} \cdot p_0 + \omega^{n_1} \cdot p_1 + \cdots + \omega^{n_k} \cdot p_k$, where $n_0 > n_1 > \cdots > n_k$
- The degree α of a Muller automaton $\mathcal{M}$ in the Wagner hierarchy is the maximal "tree of cycles" $\mathcal{T}_\alpha$ in $\mathcal{M}$



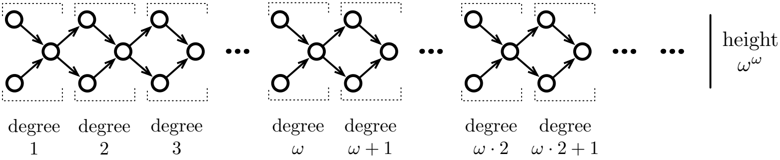
The Wagner Hierarchy – Summary

- A quasi well-ordering of transfinite height ω^ω
- Every ordinal $\alpha < \omega^\omega$ has a unique Cantor normal form $\alpha = \omega^{n_0} \cdot p_0 + \omega^{n_1} \cdot p_1 + \cdots + \omega^{n_k} \cdot p_k$, where $n_0 > n_1 > \cdots > n_k$
- The degree α of a Muller automaton $\mathcal{M}$ in the Wagner hierarchy is the maximal "tree of cycles" $\mathcal{T}_\alpha$ in $\mathcal{M}$



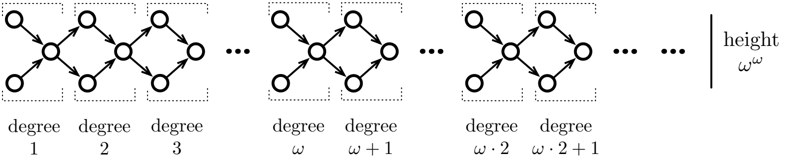
The Boolean RNNs Hierarchy

- We assume that our *Boolean RNNs* are provided with an additional specification of every of their *attractors* into a meaningful or a spurious mode
- We can transpose the Wagner hierarchy from the *Muller automata* to the *Boolean RNNs* context.
- One obtains a transfinite classification of *Boolean RNNs* according to the topological complexity of their attractors



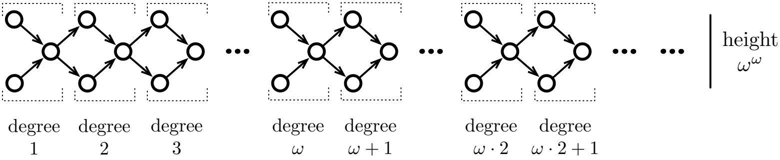
The Boolean RNNs Hierarchy

- We assume that our *Boolean RNNs* are provided with an additional specification of every of their *attractors* into a meaningful or a spurious mode
- We can transpose the Wagner hierarchy from the *Muller automata* to the *Boolean RNNs* context.
- One obtains a transfinite classification of *Boolean RNNs* according to the topological complexity of their attractors

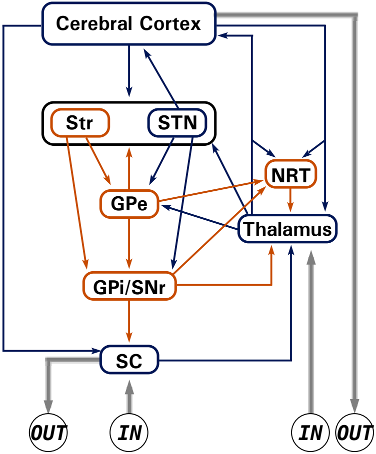


The Boolean RNNs Hierarchy

- We assume that our *Boolean RNNs* are provided with an additional specification of every of their *attractors* into a meaningful or a spurious mode
- We can transpose the Wagner hierarchy from the *Muller automata* to the *Boolean RNNs* context.
- One obtains a transfinite classification of *Boolean RNNs* according to the topological complexity of their attractors

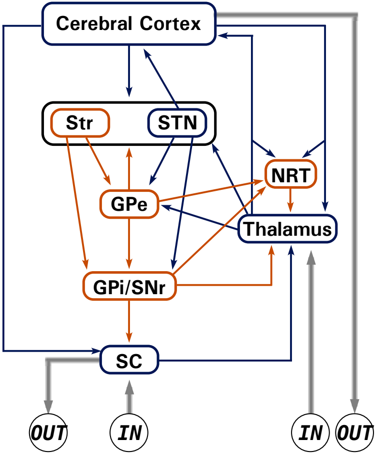


Boolean Model of the Basal-Ganglia Thalamo-cortical Network



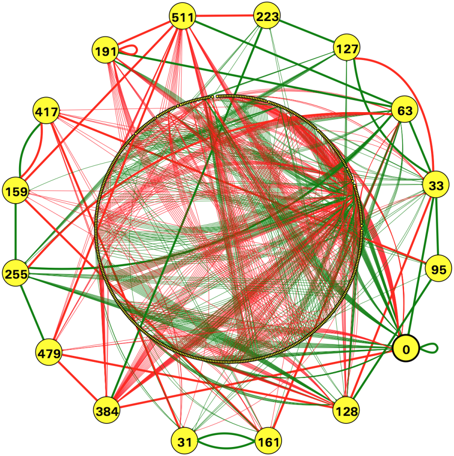
Corresponding Automaton

- 9 activation nodes and 1 input node in the network $\Rightarrow$ 512 states and a binary alphabet for the automaton



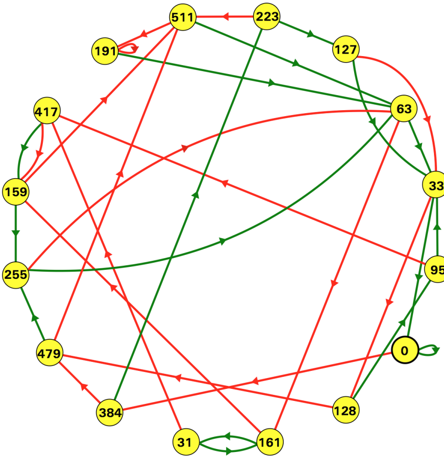
Corresponding Automaton

- 9 activation nodes and 1 input node in the network $\Rightarrow$ 512 states and a binary alphabet for the automaton



Corresponding Automaton

- 9 activation nodes and 1 input node in the network $\Rightarrow$ 512 states and a binary alphabet for the automaton



Corresponding Muller Automaton

- We provide an accepting or rejecting value to each *cycle* of the automaton, i.e., we provide an meaningful or spurious value to each *attractor* of the network
- A constitutive cycle is considered to be spurious if it is characterised either by active SC and quiet Thalamus at the same time step or by a quiet GPi/SNR during the majority of the duration of the constitutive cycle. A constitutive cycle is meaningful otherwise
- A non-constitutive cycle is considered to be meaningful if it contains a majority of meaningful constitutive cycles, and spurious otherwise

Corresponding Muller Automaton

- We provide an accepting or rejecting value to each *cycle* of the automaton, i.e., we provide an meaningful or spurious value to each *attractor* of the network
- A constitutive cycle is considered to be spurious if it is characterised either by active SC and quiet Thalamus at the same time step or by a quiet GPi/SNR during the majority of the duration of the constitutive cycle. A constitutive cycle is meaningful otherwise
- A non-constitutive cycle is considered to be meaningful if it contains a majority of meaningful constitutive cycles, and spurious otherwise

Corresponding Muller Automaton

- We provide an accepting or rejecting value to each *cycle* of the automaton, i.e., we provide an meaningful or spurious value to each *attractor* of the network
- A constitutive cycle is considered to be spurious if it is characterised either by active SC and quiet Thalamus at the same time step or by a quiet GPi/SNR during the majority of the duration of the constitutive cycle. A constitutive cycle is meaningful otherwise
- A non-constitutive cycle is considered to be meaningful if it contains a majority of meaningful constitutive cycles, and spurious otherwise

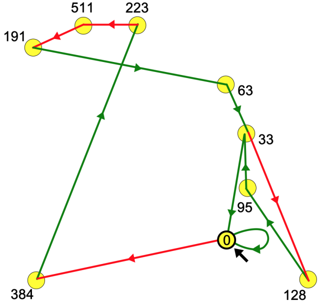
Corresponding Muller Automaton

- We provide an accepting or rejecting value to each *cycle* of the automaton, i.e., we provide an meaningful or spurious value to each *attractor* of the network
- A constitutive cycle is considered to be spurious if it is characterised either by active SC and quiet Thalamus at the same time step or by a quiet GPi/SNR during the majority of the duration of the constitutive cycle. A constitutive cycle is meaningful otherwise
- A non-constitutive cycle is considered to be meaningful if it contains a majority of meaningful constitutive cycles, and spurious otherwise

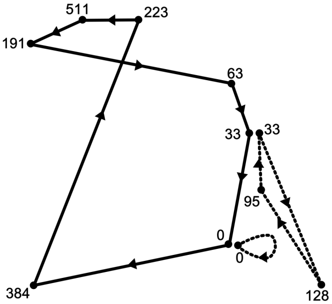
Corresponding Muller Automaton

- We provide an accepting or rejecting value to each *cycle* of the automaton, i.e., we provide an meaningful or spurious value to each *attractor* of the network
- A constitutive cycle is considered to be spurious if it is characterised either by active SC and quiet Thalamus at the same time step or by a quiet GPi/SNR during the majority of the duration of the constitutive cycle. A constitutive cycle is meaningful otherwise
- A non-constitutive cycle is considered to be meaningful if it contains a majority of meaningful constitutive cycles, and spurious otherwise

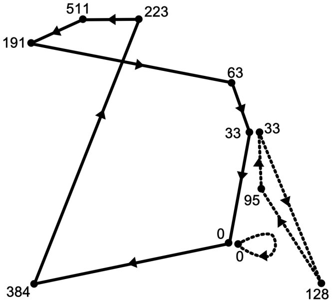
Corresponding Muller Automaton

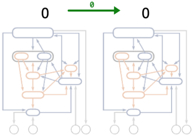


Corresponding Muller Automaton

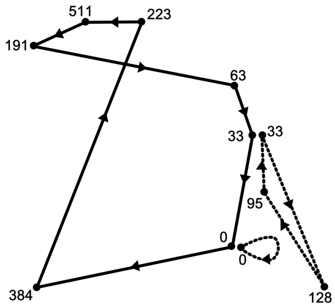


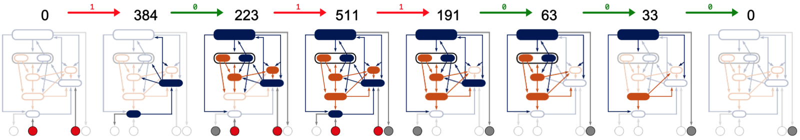
Corresponding Muller Automaton



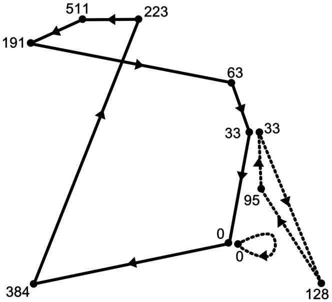


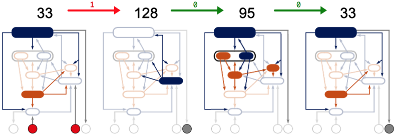
Corresponding Muller Automaton





Corresponding Muller Automaton





Attractor-Based Complexity of the Model

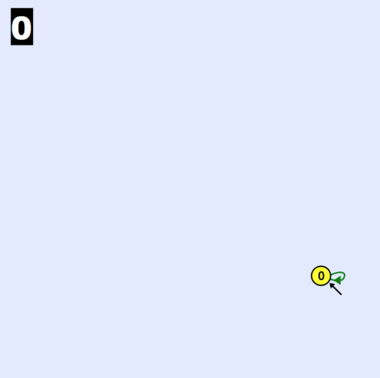
- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**

Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**

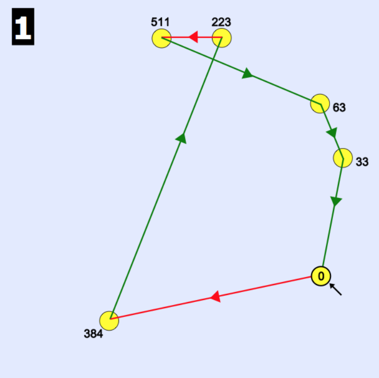
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



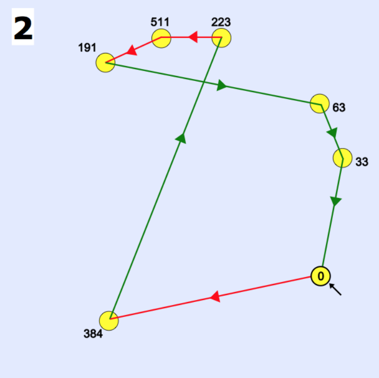
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



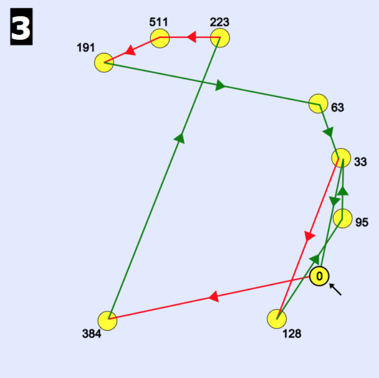
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



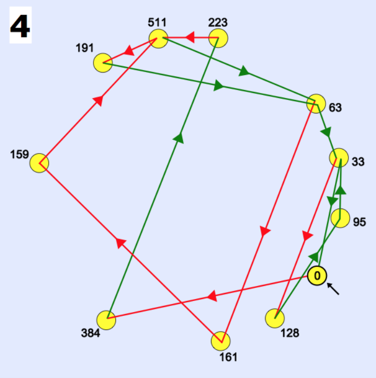
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



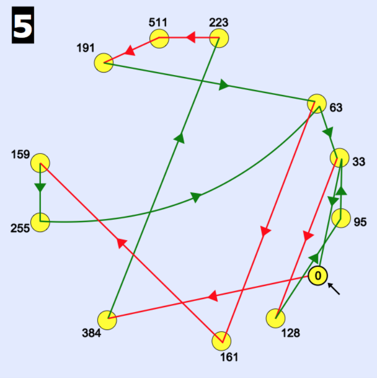
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



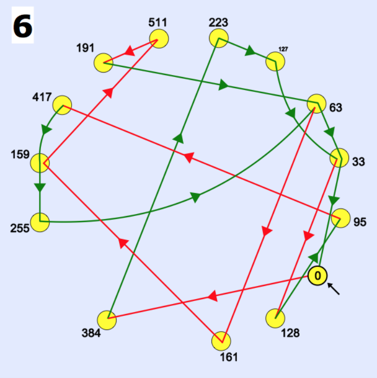
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



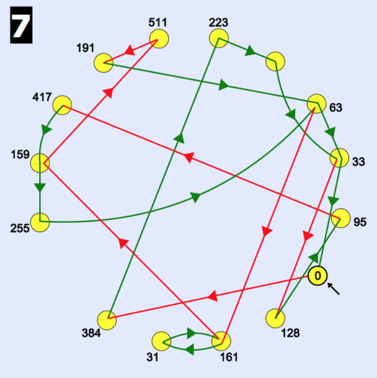
Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



Attractor-Based Complexity of the Model

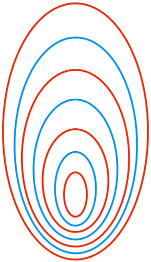
- In the corresponding Muller automaton, we found a maximal "tree of cycles" of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**



Attractor-Based Complexity of the Model

- In the corresponding Muller automaton, we found a maximal “tree of cycles” of length ω^6
- $\Rightarrow$ **Degree of the Boolean model is ω^6**

Wagner
degree



$[-]\ \omega^6$

Conclusions

- We introduced a new attractor-based complexity measurement for Boolean recurrent neural networks
- We used this complexity measure on a simplified model of the basal-ganglia thalamocortical network
- For future work, we aim to better understand the specific biological features related to this measure of complexity

Conclusions

- We introduced a new attractor-based complexity measurement for Boolean recurrent neural networks
- We used this complexity measure on a simplified model of the basal-ganglia thalamocortical network
- For future work, we aim to better understand the specific biological features related to this measure of complexity

Conclusions

- We introduced a new attractor-based complexity measurement for Boolean recurrent neural networks
- We used this complexity measure on a simplified model of the basal-ganglia thalamocortical network
- For future work, we aim to better understand the specific biological features related to this measure of complexity