Playing Muller Games in a Hurry

Joint work with John Fearnley, University of Warwick

Martin Zimmermann

RWTH Aachen University

June 18th, 2010

GandALF 2010
Minori, Italy
Motivation


*We believe that infinite games might have an interest for casual living-room recreation.*
Motivation


We believe that infinite games might have an interest for casual living-room recreation.

McNaughton suggests a method of keeping score to declare a winner such that

.. if the play were to continue with each [player] playing forever as he has so far, then the player declared to be the winner would be the winner of the infinite play of the game.
Motivation


*We believe that infinite games might have an interest for casual living-room recreation.*

McNaughton suggests a method of keeping score to declare a winner such that

.. *if the play were to continue with each [player] playing forever as he has so far, then the player declared to be the winner would be the winner of the infinite play of the game.*

“*Winning regions should be equal*”
A first idea

Consider an infinite game $G$ played on a finite graph.

- Stop a play as soon as a cycle is closed. The winner of the induced infinite play is declared to win the finite play.
- If $G$ is positionally determined, then the winning regions of both games coincide.
A first idea

Consider an infinite game $G$ played on a finite graph.

- Stop a play as soon as a cycle is closed. The winner of the induced infinite play is declared to win the finite play.
- If $G$ is positionally determined, then the winning regions of both games coincide.
- This can be extended to games $G$ that are determined with finite-state strategies: wait for a repetition of a memory state (for some fixed memory structure).
A first idea

Consider an infinite game $\mathcal{G}$ played on a finite graph.

- Stop a play as soon as a cycle is closed. The winner of the induced infinite play is declared to win the finite play.
- If $\mathcal{G}$ is positionally determined, then the winning regions of both games coincide.
- This can be extended to games $\mathcal{G}$ that are determined with finite-state strategies: wait for a repetition of a memory state (for some fixed memory structure).

Drawbacks (assuming $\mathcal{G}$ is a Muller game with $n$ vertices):

- maximal play length: $n!$.
- need to remember $n!$ memory states.

Our goal: improve both bounds.
Outline

1. Muller Games and Scoring Functions

2. Finite-time Muller Games

3. Conclusion
Muller Games

- Arena: $G = (V, V_0, V_1, E)$ with finite, directed graph $(V, E)$, partition $(V_0, V_1)$ of $V$. 
Muller Games

- Arena: $G = (V, V_0, V_1, E)$ with finite, directed graph $(V, E)$, partition $(V_0, V_1)$ of $V$.

- Muller game: $(G, \mathcal{F}_0, \mathcal{F}_1)$ with partition $(\mathcal{F}_0, \mathcal{F}_1)$ of $2^V$.

- Player $i$ wins play $\rho$ iff $\text{Inf}(\rho) = \{v \mid \exists \omega j \text{ s.t. } \rho_j = v\} \in \mathcal{F}_i$. 
Muller Games

- **Arena:** $G = (V, V_0, V_1, E)$ with finite, directed graph $(V, E)$, partition $(V_0, V_1)$ of $V$.

- **Muller game:** $(G, \mathcal{F}_0, \mathcal{F}_1)$ with partition $(\mathcal{F}_0, \mathcal{F}_1)$ of $2^V$.

- Player $i$ wins play $\rho$ iff $\text{Inf}(\rho) = \{v | \exists \omega j \text{ s.t. } \rho_j = v\} \in \mathcal{F}_i$.

- Set of strategies for Player $i$: $\Pi_i$.

- Unique play started at $v$ that is played according to $\sigma \in \Pi_i$ and $\tau \in \Pi_{1-i}$: $\text{Play}(v, \sigma, \tau)$.
Muller Games

- **Arena:** $G = (V, V_0, V_1, E)$ with finite, directed graph $(V, E)$, partition $(V_0, V_1)$ of $V$.

- **Muller game:** $(G, \mathcal{F}_0, \mathcal{F}_1)$ with partition $(\mathcal{F}_0, \mathcal{F}_1)$ of $2^V$.

- Player $i$ wins play $\rho$ iff $\text{Inf}(\rho) = \{v \mid \exists \omega j \text{ s.t. } \rho_j = v\} \in \mathcal{F}_i$.

- Set of strategies for Player $i$: $\Pi_i$.

- Unique play started at $v$ that is played according to $\sigma \in \Pi_i$ and $\tau \in \Pi_{1-i}$: $\text{Play}(v, \sigma, \tau)$.

- Winning region of Player $i$:

  $$W_i = \{v \in V \mid \exists \sigma \in \Pi_i \forall \tau \in \Pi_{1-i} : \text{Play}(v, \sigma, \tau) \text{ won by Player } i\}$$
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } Occ(x_i) = F \text{ for all } i\}$$

where $Occ(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$. 
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \to \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \ldots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } Occ(x_i) = F \text{ for all } i\}$$

where $Occ(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{ k \mid \text{exist words } x_1, \ldots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

**Example:**

<table>
<thead>
<tr>
<th>w</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max \{ k \mid \text{exist words } x_1, \ldots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
For $F \subseteq V$ define $Sc_F : V^+ \to \mathbb{N}$:

$$Sc_F(w) = \max \{ k \mid \text{exist words } x_1, \ldots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

**Example:**

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{ k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Martin Zimmermann RWTH Aachen University Playing Muller Games in a Hurry
Scoring Functions

For $F \subseteq V$ define $\text{Sc}_F : V^+ \rightarrow \mathbb{N}$:

$$\text{Sc}_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}$$

where $\text{Occ}(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Sc}_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Martin Zimmermann  RWTH Aachen University  Playing Muller Games in a Hurry  6/15
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \ldots, x_k \in V^+ \text{ s.t. }$$

$x_1 \cdots x_k \text{ is suffix of } w \text{ and } Occ(x_i) = F \text{ for all } i\}$

where $Occ(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>w</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sc$_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Martin Zimmermann RWTH Aachen University Playing Muller Games in a Hurry 6/15
For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}$$

where $\text{Occ}(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

**Example:**

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
For $F \subseteq V$ define $Sc_F : V^+ \to \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \exists \text{ words } x_1, \ldots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } Occ(x_i) = F \text{ for all } i\}$$

where $Occ(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

**Example:**

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Scoring Functions**

Martin Zimmermann  
RWTH Aachen University  
Playing Muller Games in a Hurry  
6/15
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } Occ(x_i) = F \text{ for all } i\}$$

where $Occ(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
For $F \subseteq V$ define $S_{C_F}: V^+ \rightarrow \mathbb{N}$:

$$S_{C_F}(w) = \max \{ k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

**Example:**

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{C{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}$$

where $\text{Occ}(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max \{ k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Scoring Functions

For \( F \subseteq V \) define \( Sc_F : V^+ \rightarrow \mathbb{N} \):

\[
Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}
\]

where \( \text{Occ}(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\} \).

Example:

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>a</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Sc_{{a,b}} )</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}$

where $\text{Occ}(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$Sc_{{a,b,c}}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
For $F \subseteq V$ define $S_{cF}: V^{+} \rightarrow \mathbb{N}$:

$$S_{cF}(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^{+} \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } Occ(x_i) = F \text{ for all } i\}$$

where $Occ(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

**Example:**

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{c{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S_{c{a,b,c}}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i\}$$

where $\text{Occ}(w_1 \cdots w_n) = \{v \in V \mid \exists j \text{ s.t. } w_j = v\}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc{a,b}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$Sc{a,b,c}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max\{ k \mid \text{exist words } x_1, \ldots, x_k \in V^+ \text{ s.t. } x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$.

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$Sc_{{a,b,c}}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
Scoring Functions

For $F \subseteq V$ define $Sc_F : V^+ \rightarrow \mathbb{N}$:

$$Sc_F(w) = \max \{ k \mid \text{exist words } x_1, \cdots, x_k \in V^+ \text{ s.t. } \}
\quad x_1 \cdots x_k \text{ is suffix of } w \text{ and } \text{Occ}(x_i) = F \text{ for all } i \}$$

where $\text{Occ}(w_1 \cdots w_n) = \{ v \in V \mid \exists j \text{ s.t. } w_j = v \}$. 

Example:

<table>
<thead>
<tr>
<th>$w$</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>b</th>
<th>a</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>a</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Sc_{{a,b}}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$Sc_{{a,b,c}}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
Scoring Functions

For $\mathcal{F} \subseteq 2^V$ define $\text{MaxSc}_\mathcal{F} : V^+ \cup V^\omega \to \mathbb{N} \cup \{\infty\}$:

$$\text{MaxSc}_\mathcal{F}(\rho) = \max_{F \in \mathcal{F}} \max_{w \subseteq \rho} \text{Sc}_F(w)$$
Scoring Functions

For $\mathcal{F} \subseteq 2^V$ define $\text{MaxSc}_\mathcal{F} : V^+ \cup V^\omega \rightarrow \mathbb{N} \cup \{\infty\}$:

$$\text{MaxSc}_\mathcal{F}(\rho) = \max_{F \in \mathcal{F}} \max_{w \subseteq \rho} \text{Sc}_F(w)$$

Example:

$$\begin{array}{cccccccccccccc}
\text{w} & a & a & b & b & a & a & b & c & a & b & c & a & a & c \\
\hline
\text{Sc}_{\{a,b\}} & 0 & 0 & 1 & 1 & 2 & 2 & 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
\text{Sc}_{\{a,b,c\}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 & 2 \\
\end{array}$$

$\mathcal{F} = \{\{a, b\}, \{a, b, c\}\}$:

$$\text{MaxSc}_\mathcal{F}(w) = 3$$
Outline

1. Muller Games and Scoring Functions

2. Finite-time Muller Games

3. Conclusion
Lemma

Every $w \in V^* \text{ with } |w| \geq k|V|$ satisfies $\text{MaxSc}_{2V}(w) \geq k$.

“If you play long enough, some score value will be high”

Lower bound: there are words $w_k$ of length $k|V| - 1$ with $\text{MaxSc}_{2V}(w_k) < k$. 
Results about Scoring

**Lemma**

Every \( w \in V^* \) with \( |w| \geq k|V| \) satisfies \( \text{MaxSc}_{2V}(w) \geq k \).

“If you play long enough, some score value will be high”

Lower bound: there are words \( w_k \) of length \( k|V| - 1 \) with \( \text{MaxSc}_{2V}(w_k) < k \).

**Lemma (McNaughton 2000)**

Let \( k, \ell \geq 2 \), let \( F, H \subseteq V \), let \( w \in V^* \) and \( s \in V \) such that \( \text{Sc}_F(w) < k \) and \( \text{Sc}_H(w) < \ell \). If \( \text{Sc}_F(ws) = k \) and \( \text{Sc}_H(ws) = \ell \), then \( F = H \).

“At most one score value can increase at a time”
Finite-time Muller Games

- Finite-time Muller game: \((G, \mathcal{F}_0, \mathcal{F}_1, k)\) with threshold \(k \geq 2\).
- Play: path \(w = w_1 \cdots w_n\) with \(\text{MaxSc}_V(w_0 \cdots w_n) = k\), but \(\text{MaxSc}_V(w_1 \cdots w_{n-1}) < k\).
- Previous Lemma yields unique \(F \subseteq V\) such that \(S_{c_F}(w) = k\). Player \(i\) wins \(w\) iff \(F \in \mathcal{F}_i\).
- Strategies and winning regions defined as usual.
Finite-time Muller Games

- Finite-time Muller game: \((G, F_0, F_1, k)\) with threshold \(k \geq 2\).
- Play: path \(w = w_1 \cdots w_n\) with \(\text{MaxSc}_2^V(w_0 \cdots w_n) = k\), but \(\text{MaxSc}_2^V(w_1 \cdots w_{n-1}) < k\).
- Previous Lemma yields unique \(F \subseteq V\) such that \(\text{Sc}_F(w) = k\). Player \(i\) wins \(w\) iff \(F \in \mathcal{F}_i\).
- Strategies and winning regions defined as usual.

McNaughton considered a different definition of a finite-time Muller game: stop play when some \(\text{Sc}_F\) reaches \(|F|! + 1\).

**Theorem (McNaughton 2000)**

The winning regions in a Muller game and in McNaughton’s finite-time Muller game coincide.
Main Theorem

Theorem

The winning regions in a Muller game \((G, \mathcal{F}_0, \mathcal{F}_1)\) and in the finite-time Muller game \((G, \mathcal{F}_0, \mathcal{F}_1, 3)\) coincide.
Theorem

The winning regions in a Muller game \((G, \mathcal{F}_0, \mathcal{F}_1)\) and in the finite-time Muller game \((G, \mathcal{F}_0, \mathcal{F}_1, 3)\) coincide.

We prove a stronger statement about winning strategies in the infinite-duration Muller game, which implies the theorem.

Lemma

Player \(i\) has a strategy \(\sigma\) for a Muller game \((G, \mathcal{F}_0, \mathcal{F}_1)\) such that \(\text{MaxSc}_{\mathcal{F}_{1-i}}(\text{Play}(v, \sigma, \tau)) \leq 2\) for every \(v \in W_i\) and every \(\tau \in \Pi_{1-i}\).
What about 2?

The bound 2 in the lemma is optimal:

Player 0 has a winning strategy, but cannot avoid score values of 2 for Player 1.

\[ F_0 = \{\{1, 2, 3\}, \{1\}, \{3\}\} \]
\[ F_1 = 2^{\{1, 2, 3\}} \setminus F_0 \]

One of the plays 2112 or 2332 is consistent with every winning strategy.
What about 2?

The bound 2 in the lemma is optimal:

Player 0 has a winning strategy, but cannot avoid score values of 2 for Player 1.

\[ F_0 = \{\{1, 2, 3\}, \{1\}, \{3\}\} \]

\[ F_1 = 2^{\{1,2,3\}} \setminus F_0 \]

One of the plays 2112 or 2332 is consistent with every winning strategy.

Consequence:

To show that the finite-time Muller game with threshold 2 is equivalent, we need other proof techniques.
Outline

1. Muller Games and Scoring Functions

2. Finite-time Muller Games

3. Conclusion
Conclusion

We have presented a finite-duration version of a Muller game that is equivalent to the original game.

- Reachability game on a tree; hence, simple algorithms are available.
- Our strategies are eager: they do not spend more time in “bad” loops than they have to.
Conclusion

We have presented a finite-duration version of a Muller game that is equivalent to the original game.

- Reachability game on a tree; hence, simple algorithms are available.
- Our strategies are eager: they do not spend more time in “bad” loops than they have to.

<table>
<thead>
<tr>
<th>Threshold</th>
<th>Reduction</th>
<th>McNaughton</th>
<th>here</th>
</tr>
</thead>
<tbody>
<tr>
<td>Play Length</td>
<td>$\leq n \cdot n! + 1$</td>
<td>$</td>
<td>F</td>
</tr>
<tr>
<td>Space</td>
<td>$\mathcal{O}(n!)$</td>
<td>$\mathcal{O}((n! + 1)^n)$</td>
<td>$\mathcal{O}(3^n)$</td>
</tr>
</tbody>
</table>
Open Questions

- Is the finite-time Muller game with threshold 2 equivalent to the original Muller game?
- Given a winning strategy for a finite-time Muller game, can we turn it into a winning strategy for the Muller game?