Roadmap of the talk

- Synchronizing words
- The Road Coloring Theorem
- The Road Coloring Theorem for periodic graphs
- Application to Huffman compression
- The hybrid Černý-Road Problem
**Synchronizing words**

Synchronizing word (magic word, homing sequence, Rome word): a word $w$ such that all paths labeled $w$ terminate at the same vertex.

Synchronized automaton: automaton which has a synchronizing word.

The automaton on the right is synchronized. The word $RRR$ is synchronizing.
The solitaire game

One pebble on each state.
The solitaire game: one plays $B$
The solitaire game: one plays $R$

One plays $R$
The solitaire game: one plays $B$

One plays $B$
A directed graph with constant out-degree is **road colorable** if there is a coloring of its edges such that

- the edges going out of a vertex have distinct colors;
- there is a synchronizing word.

A graph is **aperiodic** if it is strongly connected and the \( \text{gcd of the cycle lengths is 1} \).

A strongly connected graph which has a synchronized coloring is aperiodic.

A graph is **admissible** if it is strongly connected, aperiodic and has constant out-degree.

**Theorem [A. Trahtman 2007]**

Any admissible graph is road colorable.
Applications

- **lossless source coding**: the Huffman decoder can be chosen synchronized, hence resistant against errors.

- **communication protocols**: test sequences to check whether a protocol conforms to its specification.

- **symbolic dynamics**: for two aperiodic shifts of finite type $X, Y$ with the same entropy, there exists a factor map $\varphi : X \to Y$ which is almost one-to-one. (invertible on ”typical” sequences, i.e. on bi-infinite sequences which contain a synchronizing word infinitely often to the left and to the right).
A synchronizable pair of states in an automaton is a pair of states $(p, q)$ such that there is a word $w$ with $p \cdot w = q \cdot w$.

A stable pair of states in an automaton is a pair of states $(p, q)$ such that, for any word $u$, $(p \cdot u, q \cdot u)$ is a synchronizable pair.

The stable pair congruence is the relation defined on the set of states by $p \equiv q$ if $(p, q)$ is a stable pair.

**Lemma [Culik, Karhumäki, Kari, 2002]**

If the quotient of $\mathcal{A}$ by a stable pair congruence is colorable, then $\mathcal{A}$ is colorable.
Algorithm for Road Coloring

**FindColoring** (aperiodic automaton $A$, quotient automaton $B$)

1. $B \leftarrow A$
2. **while** (size($B$) > 1)
   3. **do** Update($B$)
   4. $B, (s, t) \leftarrow \text{FindStablePair}(B)$
   5. lift the coloring up from $B$ to the automaton $A$
   6. $B \leftarrow \text{Merge}(B, (s, t))$
3. return $A$
We start with some coloring and fix a color (red). The **level** of a state is its distance to the red cycle of its cluster. **Maximal states** are states of maximal level.
Finding a coloring which has a stable pair

Lemma [Trahtman 2007]
If all maximal states belong to the same tree, then there is a stable pair.
Proof of Trahtman’s lemma

Lemma [Trahtman 2007]

If all maximal states belong to the same tree, then there is a stable pair. A minimal image \( I = Q \cdot w \) is an image minimal for set inclusion. For any word \( u \), we have \( I \cdot u \) is a minimal image.

Proof.

Let \( I \) be a minimal image and \( \ell \) be the maximal level. By irreducibility, it contains a maximal state \( p \) in a tree on a cycle \( C \). If there is \( q \neq p \) maximal in \( I \), then \( |I \cdot a^\ell| < |I| \) (contradiction). Let \( m \) be a common multiple of the lengths of all red cycles. Let \( s_0 \) be the predecessor of \( r \) in \( C \) and \( s_1 \) the child of \( r \) containing \( p \). Let \( J = I \cdot a^{\ell-1} \) and \( K = J \cdot a^m \). We have \( J = \{s_1\} \cup R \) with \( R \subset 0\)-level and \( K = \{s_0\} \cup R \).
Proof of Trahtman’s lemma (2)

Lemma [Trahtman 2007]
If all maximal states belong to the same tree, then there is a stable pair.

A minimal image \( I = Q \cdot w \) is an image minimal for set inclusion. For any word \( u \), we have \( I \cdot u \) is a minimal image.

Proof.
Let \( s_0 \) be the predecessor of \( r \) in \( C \) and \( s_1 \) the child of \( r \) containing \( p \).
Let \( J = I \cdot a^{\ell-1} \) and \( K = J \cdot a^m \).
We have \( J = \{ s_1 \} \cup R \) with \( R \subseteq 0\text{-level} \) and \( K = \{ s_0 \} \cup R \).
Let \( w \) a word of minimal rank. For any word \( v \), \( |J \cdot vw| = |K \cdot vw| = |I| \).
We claim that the set \((J \cup K) \cdot vw\) is a minimal image.
Indeed, \( J \cdot vw \subseteq (J \cup K) \cdot vw \subseteq Q \cdot vw \). (all 3 are equal).
But \((J \cup K) \cdot vw = R \cdot vw \cup s_0 \cdot vw \cup s_1 \cdot vw \).
This forces \( s_0 \cdot vw = s_1 \cdot vw \).
Thus \((s_0, s_1)\) is a stable pair.
Finding a stable pair with a sequence of flips

A **flip**: an exchange of the labels (with one $a$) of two edges going out of some state.

Make a **sequence of flips** such that

- either all maximal states belong to a same tree,
- or the number $N_0$ of 0-level states increases

We consider several cases corresponding to the geometry of the automaton.
Case 1: The maximal level is zero

- If the set of outgoing edges of each state is a bunch, then there is only one red cycle, the graph is not aperiodic.
- Let $p$ with $p \xrightarrow{a} q$ and $p \xrightarrow{b} r$ and $q \neq r$. We flip these edges. We get a unique maximal tree, hence a stable pair.
Finding a stable pair with a sequence of flips (2)

Case 1: The maximal level is zero

- If the set of outgoing edges of each state is a bunch, then there is only one red cycle, the graph is not aperiodic.
- Let $p$ with $p \xrightarrow{a} q$ and $p \xrightarrow{b} r$ and $q \neq r$. We flip these edges. We get a unique maximal tree, hence a stable pair.
Case 2: The maximal level is $\ell > 0$.
Let $p$ maximal, $r$ its root, and $t \xrightarrow{b} p$.
We denote $u = t \cdot a$.

- Case 2.1. If $t$ is not in the same cluster as $r$, or if $t$ has a positive level and does not belong to the $a$-path from $p$ to $r$, we flip $t \xrightarrow{b} p$ and $t \xrightarrow{a} u$ and get an automaton which has a unique maximal tree.
Case 2: The maximal level is $\ell > 0$.
Let $p$ maximal, $r$ its root, and $t \xrightarrow{b} p$.
We denote $u = t \cdot a$.

- Case 2.2. If $t$ belongs to the $a$-path from $p$ to $r$, we flip $t \xrightarrow{b} p$ and $t \xrightarrow{a} u$ and increase the number of $N_0$ of 0-level states.
Case 2: The maximal level is $\ell > 0$.

- Case 2.3. We assume that $t$ belongs to the cycle containing $r$. Let $k_1$ be the length of the simple $a$-path from $r$ to $t$ and $k_2$ the length of the simple $a$-path from $u$ to $r$.

- If $k_2 > \ell$, we flip the edges $t \xrightarrow{b} p$ and $t \xrightarrow{a} u$ and get an automaton which has a unique maximal tree.
Case 2.3 : The maximal level is $\ell > 0$, $t \in C$.

- If $k_2 < \ell$, we flip the edges $t \xrightarrow{b} p$ and $t \xrightarrow{a} u$ and get an automaton which has strictly more states of null level since $k_1 + \ell + 1 > k_1 + k_2 + 1$. 
Finding a stable pair with a sequence of flips (7)

Case 2.3 : The maximal level is $\ell > 0$, $t \in C$.

- If $k_2 = \ell$, let $q$ be the predecessor of $r$ on the cycle, let $s$ be the child of $r$ ascendant of $p$ in the maximal tree $T$.
- If $q$ has no bunch, there are edges $q \xrightarrow{a} r$ and $q \xrightarrow{c} v$ with $v \neq r$.
  We flip these edges. If $r$ belongs to the new red cycle, $N_0$ increases.
  If not, level$(r) \geq 1$ in the new automaton and thus the new automaton has a unique maximal tree.
Case 2.3 : The maximal level is $\ell > 0$, $t \in C$.

- If $k_2 = \ell$, let $q$ be the predecessor of $r$ on the cycle, let $s$ be the child of $r$ descendant of $p$ in the maximal tree $T$.
- If $q$ and $s$ have bunches, $(q, s)$ is a stable pair.
Case 2.3: The maximal level is $\ell > 0$, $t \in C$.

- We have $k_2 = \ell$. If $q$ has a bunch and $s$ not, there are edges $s \xrightarrow{a} r$ and $s \xrightarrow{c} \nu$, with $\nu \neq r$. If there is an $a$-path from $\nu$ to $s$, we flip the two edges, creating a new red cycle, which increases $N_0$. 
Case 2.3: The maximal level is $\ell > 0$, $t \in C$.

- We have $k_2 = \ell$. If $q$ has a bunch and $s$ not. There is no $a$-path from $v$ to $s$ and $\text{level}(v) > 0$, we flip $s \xrightarrow{a} r$ and $s \xrightarrow{c} v$. 
Case 2.3 : The maximal level is $\ell > 0$, $t \in C$.

- We have $k_2 = \ell$. If $q$ has a bunch and $s$ not. We assume that $\text{level}(v) = 0$, and $v$ is in another cluster, we flip $s \xrightarrow{a} r$ and $s \xrightarrow{c} v$ and also $t \xrightarrow{a} u$ and $t \xrightarrow{b} p$. 
Case 2.3: The maximal level is $\ell > 0$, $t \in C$.

- We have $k_2 = \ell$. If $q$ has a bunch and $s$ not. We assume that $\text{level}(v) = 0$, and $v$ is in another cluster, we flip $s \xrightarrow{a} r$ and $s \xrightarrow{c} v$ and also $t \xrightarrow{a} u$ and $t \xrightarrow{b} p$. 

![Diagram](image-url)
Case 2.3.3 : The maximal level is $\ell > 0$, $t \in C$, $k_2 = \ell$

- If $q$ has a bunch and $s$ not, and there are edges $s \xrightarrow{a} r$ and $s \xrightarrow{c} v$, with $v \neq r$ in $C$. Let us denote by $k_3$ the length of the simple $a$-path from $u$ to $v$.

Since $v \neq r$, we have $k_3 \neq k_2$. Hence $k_3 < \ell$ or $k_3 > \ell$. We flip the edges $s \xrightarrow{a} r$ and $s \xrightarrow{c} v$ and proceed as in Case 2.3.1 or 2.3.2.
Complexity of the Road Coloring problem

The complexity of Trathman’s algorithm is cubic $O(\text{card } A \times n^3)$.

Theorem [B., Perrin 2008 preprint]

One can compute a synchronized coloring of an $n$-state admissible graph in time $O(\text{card } A \times n^2)$.
The period of a graph is the gcd of the lengths of the cycles. The rank of a colored graph \((Q, E)\) is the minimal cardinality of the sets \(Q \cdot u\) for all colored sequences \(u\).

Theorem [B., Perrin preprint 2008, Budzban, Feinsilver 2011]

Any strongly connected with constant outgoing arity has a coloring whose rank is equal to the period of the graph.
Application to Huffman compression

\[ \begin{align*}
  a & : \frac{1}{16}, \quad b : \frac{1}{16}, \quad c : \frac{1}{8}, \quad d : \frac{1}{16}, \quad e : \frac{1}{16}, \\
  f & : \frac{1}{8}, \quad g : \frac{1}{8}, \quad h : \frac{1}{8}, \quad i : \frac{1}{4} \\
\end{align*} \]

Encoding: \( d \leftrightarrow \text{RBRR} \)
A non synchronized Huffman decoder.

When the lengths of the codewords in are relatively prime, there is another Huffman code (with the same length distribution) with a synchronized decoder.
Application to Huffman compression (3)
Application to Huffman compression (3)
Quotient + flip at 356
Lifting up the flips
Lifting up the flips

The sequence \textbf{RBR} is a homing sequence.

The Road Coloring algorithm makes the Huffman decoder synchronized.
Volkov raised the following question:

What is the minimum length of a synchronizing word for a synchronized coloring of an aperiodic graph?

We conjecture that a synchronized coloring such that the coloring is moreover one-cluster can be obtained with the same complexity. This guarantees a minimum length of a synchronizing word of a length at most quadratic in the number of vertices.