# Zeta function and Entropy of Visibly Pushdown Systems

Marie-Pierre Béal, Michel Blockelet, Cătălin Dima, Pavel Heller

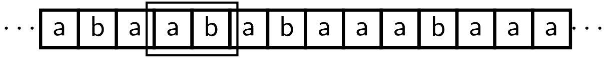
Université Paris-Est
Laboratoire d'informatique Gaspard-Monge UMR 8049

ANR EQINOCS
Paris, May 2016

# Overview

- Background : Shifts of finite type. Sofic shifts
- Zeta functions of shifts
- Dyck shifts and visibly-pushdown shifts
- Entropy

# Shifts of sequences

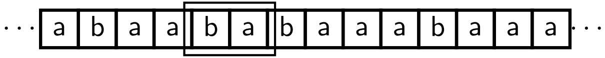



$A$ is a finite alphabet

$F$ is a set of finite words over $A$ (forbidden patterns or factors)

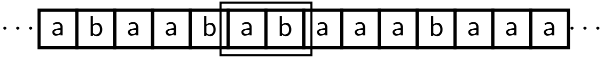
$X_F$: the subset of $A^{\mathbb{Z}}$ of sequences of letters avoiding $F$.

# Shifts of sequences




$A$ is a finite alphabet
$F$ is a set of finite words over $A$ (forbidden patterns or factors)
$X_F$: the subset of $A^{\mathbb{Z}}$ of sequences of letters avoiding $F$.

# Shifts of sequences



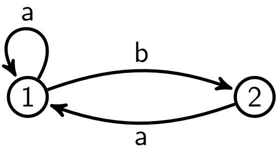

$A$ is a finite alphabet
$F$ is a set of finite words over $A$ (forbidden patterns or factors)
$\mathsf{X}_F$: the subset of $A^{\mathbb{Z}}$ of sequences of letters avoiding $F$.

# Shifts of finite type

A forbidden sequence:

$$\cdots \text{ abaababababaa}\textcolor{red}{\text{bb}}\text{aaababababa } \cdots$$

Characterized by a finite set of forbidden blocks $F = \{bb\}$.

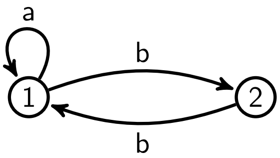


# Sofic shifts

A forbidden sequence:

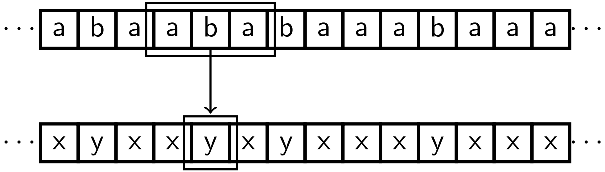
$$\cdots\ \text{abbaabbabbabbaaab}\textcolor{red}{\text{abbba}}\text{aaabbabbaaa}\ \cdots$$

Characterized by a regular set of forbidden patterns: an odd number of *b* between two *a* is forbidden.



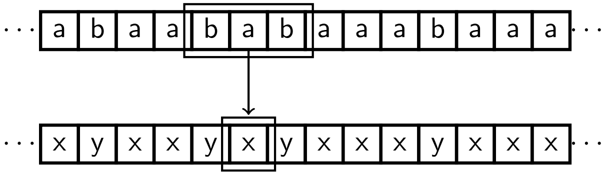
# Conjugacy between shifts

A one-to-one and onto sliding block code $\Phi : X \subseteq A^{\mathbb{Z}} \to Y \subseteq B^{\mathbb{Z}}$.
The inverse is also a sliding block code.



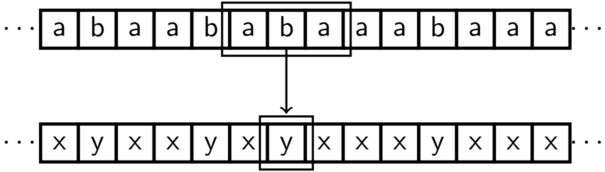
# Conjugacy between shifts

A one-to-one and onto sliding block code $\Phi : X \subseteq A^{\mathbb{Z}} \to Y \subseteq B^{\mathbb{Z}}$.
The inverse is also a sliding block code.



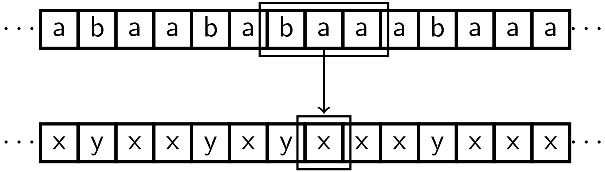
# Conjugacy between shifts

A one-to-one and onto sliding block code $\Phi : X \subseteq A^{\mathbb{Z}} \to Y \subseteq B^{\mathbb{Z}}$.
The inverse is also a sliding block code.



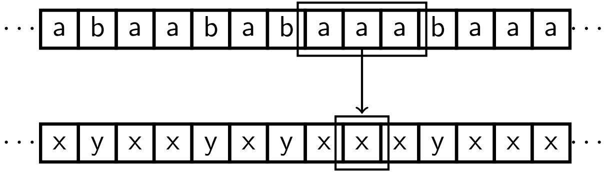
# Conjugacy between shifts

A one-to-one and onto sliding block code $\Phi : X \subseteq A^{\mathbb{Z}} \to Y \subseteq B^{\mathbb{Z}}$.
The inverse is also a sliding block code.



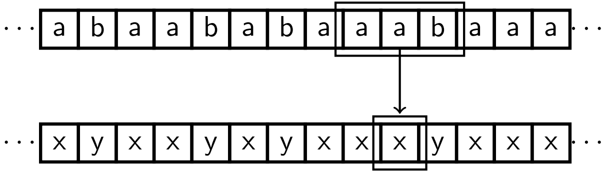
# Conjugacy between shifts

A one-to-one and onto sliding block code $\Phi : X \subseteq A^{\mathbb{Z}} \to Y \subseteq B^{\mathbb{Z}}$.
The inverse is also a sliding block code.

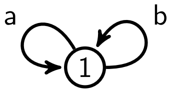


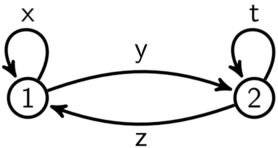
# Conjugacy between shifts

A one-to-one and onto sliding block code $\Phi : X \subseteq A^{\mathbb{Z}} \to Y \subseteq B^{\mathbb{Z}}$.
The inverse is also a sliding block code.



# Conjugate shifts: example




$$A = \begin{bmatrix} 2 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

It is not known

- if it is decidable whether two shifts of finite type are conjugate.

# Zeta function: counting periodic sequences

$(X, \sigma)$ is a shift with $\sigma : (x_i)_{i\in\mathbb{Z}} \to (x_{i+1})_{i\in\mathbb{Z}}$
$p_n$ is the number of sequences $x \in X$ such that $\sigma^n(x) = x$

The zeta function of $X$ is defined as

$$\zeta_X(z) = \exp \sum_{n\geq 1} \frac{p_n}{n} z^n = \prod_{\gamma \text{ periodic orbit}} (1 - z^{|\gamma|})^{-1}.$$

Periodic pattern abaaba

$$\cdots \text{abaaba abaaba abaaba abaaba} \cdots$$

Note that $\frac{\mathrm{d}}{\mathrm{dz}} \log \zeta_X(z) = \sum_{n\geq 1} p_n z^n$

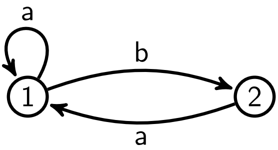
# A simple example

$X = \{a, b\}^{\mathbb{Z}}$.

$$\begin{aligned}
\zeta_X(z) &= \exp \sum_{n \geq 1} \frac{p_n}{n} z^n \\
&= \exp \sum_{n \geq 1} \frac{2^n}{n} z^n \\
&= \exp \sum_{n \geq 1} \frac{(2z)^n}{n} \\
&= \exp \log \frac{1}{1 - 2z} \\
&= \frac{1}{1 - 2z} = (2z)^*
\end{aligned}$$

# Zeta function of shifts of finite type

Bowen and Lanford 1970




$$\mathcal{A} = (Q, E) \qquad A = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \qquad \zeta_X(z) = \frac{1}{1 - z - z^2} = (z + z^2)^*$$

**Theorem (Bowen and Lanford 1970)**

*If $X$ is a shift of finite type,*

$$\zeta_X(z) = \frac{1}{\det(I - Az)}$$

# Zeta function of sofic shifts

Manning 1971, Bowen 1978
$\mathcal{A} = (Q, E)$ $Q = \{p_1 < p_2 < \cdots < p_n\}$.
$\mathcal{A}_{\otimes k} = (Q_{\otimes k}, E_{\otimes k})$, where $Q_{\otimes k}$is the set of all ordered $k$-uples of states of $Q$, and the edge are

$$(p_1, \ldots, p_k) \xrightarrow{a} (q'_1, \ldots, q'_k) \text{ iff } \begin{cases} p_i \xrightarrow{a} q_i \text{ in } \mathcal{A} \\ (q'_1, \ldots, q'_k) = \pi_{even}(q_1, \ldots, q_k) \end{cases}$$

$$(p_1, \ldots, p_k) \xrightarrow{-a} (q'_1, \ldots, q'_k) \text{ iff } \begin{cases} p_i \xrightarrow{a} q_i \text{ in } \mathcal{A} \\ (q'_1, \ldots, q'_k) = \pi_{odd}(q_1, \ldots, q_k) \end{cases}$$

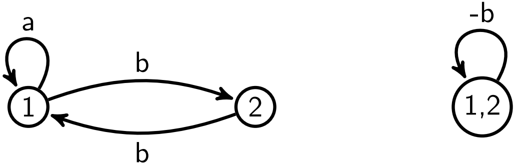
**Theorem (Bowen 1978)**

*If $X$ is a sofic shift,*

$$\zeta_X(z) = \prod_{\ell=1}^{|Q|} \det(I - A_{\otimes \ell}(z))^{(-1)^\ell}$$

# Zeta function of sofic shifts

Manning 1971, Bowen 1978




$$\zeta_X(z) = \frac{\det(I - A_{\otimes 2} z)}{\det(I - Az)} = \frac{1+z}{1-z-z^2} = (1+z)(z+z^2)^*$$

# Multivariate zeta functions

Berstel and Reutenauer 1990
$P(X)$ is the (non commutative) formal series of periodic patterns of $X$.
The *multivariate zeta function* of $X$ is the commutative series in $\mathbb{Z}[\![A]\!]$

$$Z(X) = \exp \sum_{n \geq 1} \frac{\underline{\underline{[P(X)]_n}}}{n},$$

where each $[P(X)]_n$ is the homogeneous part of $P(X)$ of degree $n$.

$$\zeta_X(z) = \theta(Z(X)),$$

where $\theta(a) = z$ for any letter $a \in A$.

# $\mathbb{N}$-rationality of zeta functions of sofic shifts

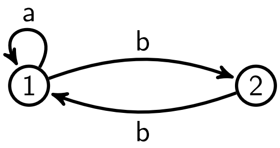
Reutenauer 1997

**Theorem (Reutenauer 1997)**

*Let $X$ be a sofic shift. There is a finite rational factorization $(\mathcal{C}_i)_{i\in I}$ of $A^*$ such that*

$$Z(X) = \prod_{j\in J\subseteq I} \mathcal{C}_j^*$$

If $(\mathcal{C}_i)_{i\in I}$ is a factorization then each set $\mathcal{C}_i$ is a circular code and each conjugacy class of nonempty words meets exactly one $\mathcal{C}_i^*$

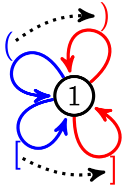



$$Z(X) = b^*(a(bb)^*)^* = \mathcal{C}_2^*\mathcal{C}_1^* (= \frac{1+b}{1-a-bb})$$

# Beyond sofic constraints: the Dyck shift

Krieger 1974

$A = (A_c, A_r)$ call alphabet $\{(, [\}$ return alphabet $\{), ]\}$
Dyck($A$) language generated by the grammar $X \to cXrX \mid \varepsilon$
The Dyck shift is $\mathsf{X}_F$ where $F = $ "(" Dyck($A$) "]" $\cup$ "[" Dyck($A$) ")"
Allowed factors are factors of well-parenthesized words




An allowed sequence: $\cdots ) ) ) ] ( ( ) ) ] [ ] [ ( \cdots$.

# Zeta function of the Dyck shift

Keller 1991
A set of words $\mathcal{C}$ such that each bi-infinite sequence has at most one decomposition into words of $\mathcal{C}$ is a circular code.

Let $\mathcal{A} = (Q, E)$ be a directed labeled graph over $A$
$(\mathcal{A}, \mathcal{C})$ is a circular Markov code if each bi-infinite label of a path of $\mathcal{A}$ has at most one decomposition into words of $\mathcal{C}$.

$C_{pq} = A^*_{pq} \cap \mathcal{C}$.
$\mathsf{X}_C$ is the $\sigma$-invariant set of orbits of the bi-infinite sequences $(e_i)$ with $e_i \in C_{p_i p_{i+1}}$.

**Theorem (Keller 1991)**

*Let $(\mathcal{A}, \mathcal{C})$ be a circular Markov code.*

$$\zeta_{\mathsf{X}_C}(z) = \frac{1}{\det(I - C(z))}$$

# Zeta function of the Dyck shift

Encoding of periodic patterns of the Dyck shift $X$.

Dyck($X$): the set of well-parenthesized blocks of $X$: $\varepsilon$, ( ), [ ], ( [ ] ) ( ), ...

$C = \mathrm{Prime}(X) = \mathrm{Dyck}(X) - (\mathrm{Dyck}(X))^2$
the set of prime Dyck words of $X$

$A = (A_c, A_r)$ call alphabet $\{(, [\}$ return alphabet $\{), ]\}$
$C$, $C(A_r)^*$, $(A_c)^*C$, $A_c$, $A_r$ are circular codes

**Theorem (Keller 1991)**

*Let $X$ be the Dyck shift over $2N$ symbols*

$$\zeta_X(z) = \frac{\zeta_{X_{(A_c)^*C}}(z)\zeta_{X_{C(A_r)^*}}(z)\zeta_{X_{A_r}}(z)\zeta_{X_{A_c}}(z)}{\zeta_{X_C}(z)}$$

# Zeta function of the Dyck shift

Encoding of periodic patterns of the Dyck shift $X$.

Dyck($X$): the set of well-parenthesized blocks of $X$: $\varepsilon$, ( ), [ ], ( [ ] ) ( ), ...

$C = \text{Prime}(X) = \text{Dyck}(X) - (\text{Dyck}(X))^2$
the set of prime Dyck words of $X$

$A = (A_c, A_r)$ call alphabet $\{(, [\}$ return alphabet $\{), ]\}$
$C$, $C(A_r)^*$, $(A_c)^*C$, $A_c$, $A_r$ are circular codes

**Theorem (Keller 1991)**

*Let $X$ be the Dyck shift over $2N$ symbols*

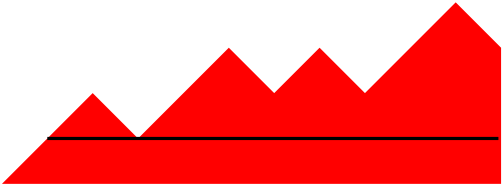
$$\zeta_X(z) = \frac{\det(I - C)}{\det(I - A_c^* C)\det(I - CA_r^*)\det(I - A_r)\det(I - A_c)}$$

$$= \frac{2(1 + \sqrt{1 - 4Nz^2})}{(1 - 2Nz + \sqrt{1 - 4Nz^2})^2}$$

# Zeta function of the Dyck shift

Encoding of periodic sequences. Case balance$(w) > 0$

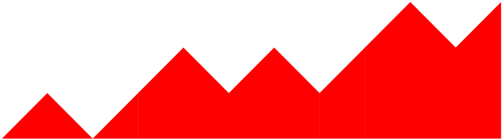
$w = aabaababaaba$



# Zeta function of the Dyck shift

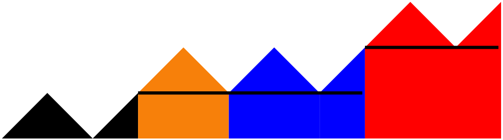
Encoding of periodic sequences. Case balance($w$) $> 0$

$u = abaababaabaa$



# Zeta function of the Dyck shift

Encoding of periodic sequences. Case balance$(w) > 0$
$u = abaababaabaa \in (CA_c{}^*)^*$



# Zeta function of Markov-Dyck shift

Krieger and Matsumoto 2011

$G = (Q, E)$ be a directed multigraph
$G^- = (Q, E^-)$, $E^-$ a copy of $E$
$G^+ = (Q, E^+)$ reversed graph

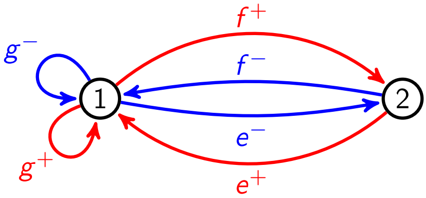
Graph inverse semigroup $S$: the semigroup generated by $Q \cup E^- \cup E^+$ with a zero quotiented by

- $pq = 0$ if $p \neq q$ and $p^2 = p$
- $e^- f^+ = 0$ if $f \neq e$
- $e^- e^+ = i(e)$
- $i(e)e^- = e^- t(e)$, $t(e)e^+ = e^+ s(e)$

The shift $X(G)$ is the set of bi-infinite paths of $G^- \cup G^+$ with no factor zero in $S$

# Zeta function of Markov-Dyck shifts

Krieger and Matsumoto 2011




An allowed sequence: $\cdots e^- f^- f^+ e^+ g^- g^+ g^+ g^+ \cdots$.

$(C_{pq})_{p,q \in Q}$: $C_{pp}$ is the set of prime paths from $p$ to $p$ of value $s(p)$ in $S$

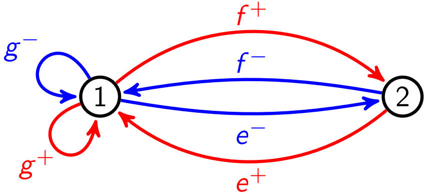
**Theorem (Krieger Matsumoto 2011)**

*Let $X$ be a Markov-Dyck shift.*

$$\zeta_X(z) = \frac{\zeta_{X_{(M_-)^* C}}(z)\,\zeta_{X_{C(M_+)^*}}(z)\,\zeta_{X_{M_+}}(z)\,\zeta_{X_{M_-}}(z)}{\zeta_{X_C}(z)}$$

# Zeta function of Markov-Dyck shifts

Krieger and Matsumoto 2011




An allowed sequence: $\cdots e^- f^- f^+ e^+ g^- g^+ g^+ g^+ \cdots$.

$(C_{pq})_{p,q \in Q}$: $C_{pp}$ is the set of prime paths from $p$ to $p$ of value $s(p)$ in $S$

**Theorem (Krieger Matsumoto 2011)**

*Let $X$ be the Dyck shift over $2N$ symbols*

$$\zeta_X(z) = \frac{\det(I - C)}{\det(I - M_-^* C)\det(I - CM_+^*)\det(I - M_+)\det(I - M_-)}$$

# Sofic-Dyck (or visibly pushdown shifts)

$A = (A_c, A_r, A_i)$ call, return and internal (or neutral) alphabet.
Dyck($A$): words where each call symbol is matched with a return one
Dyck automaton $\mathcal{A} = (G, M)$
$G = (Q, E)$ is a directed labeled multigraph
$M$ is a set of pairs of matched edges.

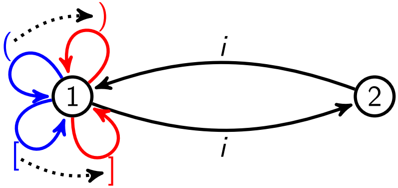
A finite path $\pi$ is admissible if for any factor of $\pi$

$$p \xrightarrow{c} q \overbrace{\rightarrow \cdots \rightarrow}^{\pi_1} p' \xrightarrow{r} q'$$

where label$(\pi_1) \in$ Dyck($A$), then $p \xrightarrow{c} q$ and $p' \xrightarrow{r} q'$ are matched.

An infinite path is admissible if all its finite factors are admissible.
$\mathsf{X}_{\mathcal{A}}$ is the set of labels of bi-infinite admissible paths of $\mathcal{A}$.

# Sofic-Dyck shifts




An allowed sequence: $\cdots ( ( i \, i ) ) [ i \, i ] ( ( \cdots$.

**Theorem (Béal, Blockelet, Dima 2014)**

*Sofic-Dyck shifts over $A$ are the exactly the shifts $X_F$ where $F$ is a visibly pushdown language over $A$.*

# Visibly pushdown languages

Mehlhorn 1980 Input-driven languages
Alur and Madhusudan 2004
$M = (Q, I, \Gamma, \Delta, F)$

- $Q$ is the finite state of states
- $A = (A_c, A_r, A_i)$ is the partitioned alphabet
- $\Gamma$ is the stack alphabet
- $\Delta \subset \begin{cases} Q \times A_c \times Q \times (\Gamma \setminus \{\bot\}) \\ Q \times A_r \times (\Gamma \setminus \{\bot\}) \times Q \\ Q \times A_i \times Q \end{cases}$

$(p, \ell, q) \in \Delta \qquad p, \left|\begin{array}{c} \alpha \\ \vdots \\ \beta \\ \bot \end{array}\right|$

# Visibly pushdown languages

Mehlhorn 1980 Input-driven languages
Alur and Madhusudan 2004
$M = (Q, I, \Gamma, \Delta, F)$

- $Q$ is the finite state of states
- $A = (A_c, A_r, A_i)$ is the partitioned alphabet
- $\Gamma$ is the stack alphabet
- $\Delta \subset \begin{cases} Q \times A_c \times Q \times (\Gamma \setminus \{\bot\}) \\ Q \times A_r \times (\Gamma \setminus \{\bot\}) \times Q \\ Q \times A_i \times Q \end{cases}$

$$(p, \ell, q) \in \Delta \qquad p, \left|\begin{array}{c} \alpha \\ \vdots \\ \beta \\ \bot \end{array}\right| \xrightarrow{\ \ell\ } q, \left|\begin{array}{c} \alpha \\ \vdots \\ \beta \\ \bot \end{array}\right|$$

# Visibly pushdown languages

Mehlhorn 1980 Input-driven languages
Alur and Madhusudan 2004
$M = (Q, I, \Gamma, \Delta, F)$

- $Q$ is the finite state of states
- $A = (A_c, A_r, A_i)$ is the partitioned alphabet
- $\Gamma$ is the stack alphabet
- $\Delta \subset \begin{cases} Q \times A_c \times Q \times (\Gamma \setminus \{\bot\}) \\ Q \times A_r \times (\Gamma \setminus \{\bot\}) \times Q \\ Q \times A_i \times Q \end{cases}$

$(p, a, q, \alpha) \in \Delta$ $\qquad p, \begin{array}{|c|} \alpha \\ \vdots \\ \beta \\ \bot \\ \hline \end{array}$

# Visibly pushdown languages

Mehlhorn 1980 Input-driven languages

Alur and Madhusudan 2004

$M = (Q, I, \Gamma, \Delta, F)$

- $Q$ is the finite state of states
- $A = (A_c, A_r, A_i)$ is the partitioned alphabet
- $\Gamma$ is the stack alphabet
- $\Delta \subset \begin{cases} Q \times A_c \times Q \times (\Gamma \setminus \{\bot\}) \\ Q \times A_r \times (\Gamma \setminus \{\bot\}) \times Q \\ Q \times A_i \times Q \end{cases}$

$(p, a, q, \alpha) \in \Delta$

$$p, \left|\begin{array}{c} \alpha \\ \vdots \\ \beta \\ \bot \end{array}\right| \xrightarrow{a} q, \left|\begin{array}{c} \alpha \\ \alpha \\ \vdots \\ \beta \\ \bot \end{array}\right|$$

# Visibly pushdown languages

Mehlhorn 1980 Input-driven languages
Alur and Madhusudan 2004
$M = (Q, I, \Gamma, \Delta, F)$

- $Q$ is the finite state of states
- $A = (A_c, A_r, A_i)$ is the partitioned alphabet
- $\Gamma$ is the stack alphabet
- $\Delta \subset \begin{cases} Q \times A_c \times Q \times (\Gamma \setminus \{\bot\}) \\ Q \times A_r \times (\Gamma \setminus \{\bot\}) \times Q \\ Q \times A_i \times Q \end{cases}$

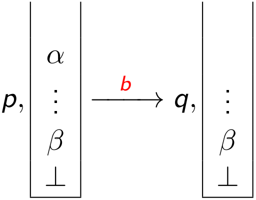
$(p, b, \alpha, q) \in \Delta \qquad p, \begin{array}{|c|} \alpha \\ \vdots \\ \beta \\ \bot \\ \hline \end{array}$

# Visibly pushdown languages

Mehlhorn 1980 Input-driven languages
Alur and Madhusudan 2004
$M = (Q, I, \Gamma, \Delta, F)$

- $Q$ is the finite state of states
- $A = (A_c, A_r, A_i)$ is the partitioned alphabet
- $\Gamma$ is the stack alphabet
- $\Delta \subset \begin{cases} Q \times A_c \times Q \times (\Gamma \setminus \{\bot\}) \\ Q \times A_r \times (\Gamma \setminus \{\bot\}) \times Q \\ Q \times A_i \times Q \end{cases}$

$(p, b, \alpha, q) \in \Delta$



# Zeta function of sofic-Dyck shifts

Béal, Blockelet, Dima 2014 with a Keller-like encoding of periodic patterns
Béal, Dima, Heller 2015 with a new encoding of periodic patterns

$A = (A_c, A_r, A_i)$
Dyck automata $\mathcal{A} = (G, M)$ left-reduced (resp. $\mathcal{A}'$ right-reduced)

$C = (C_{pq})$, where $C_{pq}$ is the set of prime Dyck words labeling an admissible path from $p$ to $q$
$M_c = (M_{c,pq})$, (resp. $M_r$) where $M_{c,pq}$ is the sum of call (resp. return) letters labeling an edge from $p$ to $q$

**Proposition (A new encoding of periodic patterns)**

*Let $X$ be a sofic-Dyck shift, $\mathcal{P}(X)$ the set of periodic patterns of $X$*

$$\mathcal{P}(X) = \mathcal{P}(\mathsf{X}_{C^* M_c}) \sqcup \mathcal{P}(\mathsf{X}_{M_r + C})$$

# Zeta function of sofic-Dyck shifts

Béal, Blockelet, Dima 2014 with a Keller-like encoding of periodic patterns
Béal, Dima, Heller 2015 with a new encoding of periodic patterns

$A = (A_c, A_r, A_i)$
Dyck automata $\mathcal{A} = (G, M)$ left-reduced (resp. $\mathcal{A}'$ right-reduced)

$C = (C_{pq})$, where $C_{pq}$ is the set of prime Dyck words labeling an admissible path from $p$ to $q$
$M_c = (M_{c,pq})$, (resp. $M_r$) where $M_{c,pq}$ is the sum of call (resp. return) letters labeling an edge from $p$ to $q$

**Proposition**

*Let $X$ be a the sofic-Dyck shift.*

$$Z(X) = Z(\mathsf{X}_{C^* M_c}) Z(\mathsf{X}_{M_r + C})$$

# Zeta function of sofic-Dyck shifts

Béal, Blockelet, Dima 2014 with a Keller-like encoding of periodic patterns
Béal, Dima, Heller 2015 with a new encoding of periodic patterns

$A = (A_c, A_r, A_i)$
Dyck automata $\mathcal{A} = (G, M)$ left-reduced (resp. $\mathcal{A}'$ right-reduced)

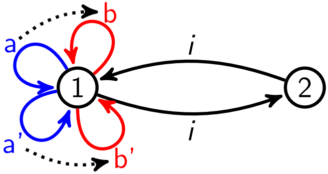
$C = (C_{pq})$, where $C_{pq}$ is the set of prime Dyck words labeling an admissible path from $p$ to $q$
$M_c = (M_{c,pq})$, (resp. $M_r$) where $M_{c,pq}$ is the sum of call (resp. return) letters labeling an edge from $p$ to $q$

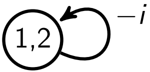
**Theorem**

*Let $X$ be a the sofic-Dyck shift.*

$$Z(X) = \prod_{\ell=1}^{|Q|} \det(I - (C^* M_c)_{\otimes \ell})^{(-1)^\ell} \prod_{\ell=1}^{|Q'|} \det(I - (C' + M_r')_{\otimes \ell})^{(-1)^\ell}.$$

# Example



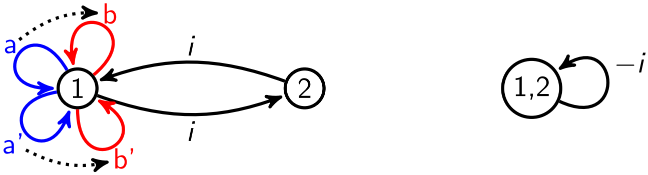




$$C_{11} = aD_{11}b + a'D_{11}b'$$

$$D_{11} = aD_{11}bD_{11} + a'D_{11}b'D_{11} + iiD_{11} + \varepsilon$$

$$Z(X) = \frac{(1+i)}{(1-(C_{11}+i^2)^*(a+a'))(1-(C_{11}+i^2+b+b'))}$$

# Example




$$\zeta_X(z) = \frac{(1+z)(1-z^2-\frac{1-z^2-\sqrt{1-10z^2+z^4}}{2})}{(1-2z-z^2-\frac{1-z^2-\sqrt{1-10z^2+z^4}}{2})^2}$$

$$h(X) = \log\frac{1}{\rho} = \log\frac{2}{\sqrt{13}-3} \sim \log 3.3027.$$

# $\mathbb{N}$-algebraicity of the zeta function of sofic-Dyck shifts

Using Reutenauer's result

**Theorem (Béal, Dima, Heller 2015)**

*Let $X$ be sofic-Dyck shift. There is a finite number of visibly pushdown circular codes $(C_j)_{j \in J}$ such that*

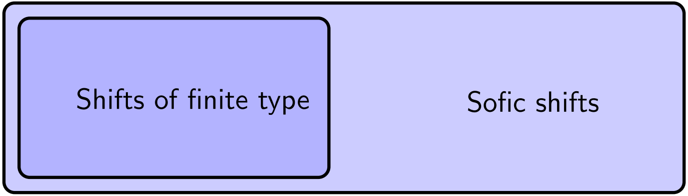
$$Z(X) = \prod_{j \in J} C_j^*$$

.

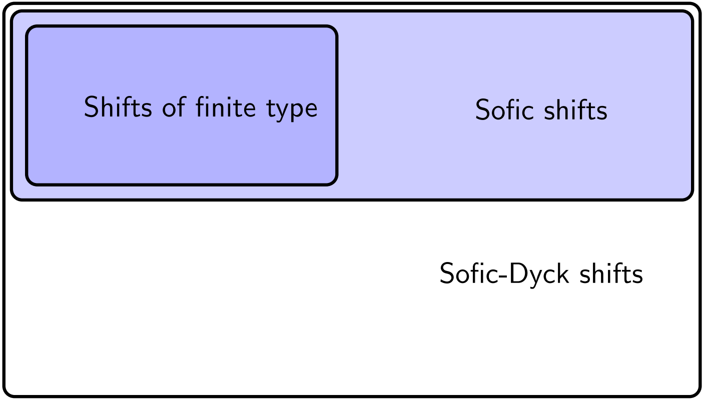
$Z(X)$ is the (commutative image of) the generating series of a visibly pushdown language

$$Z(X) = (C_{11} + i^2)^*(a + a'))^* i^*(C_{11} + b + b')(i^2)^*)^*$$

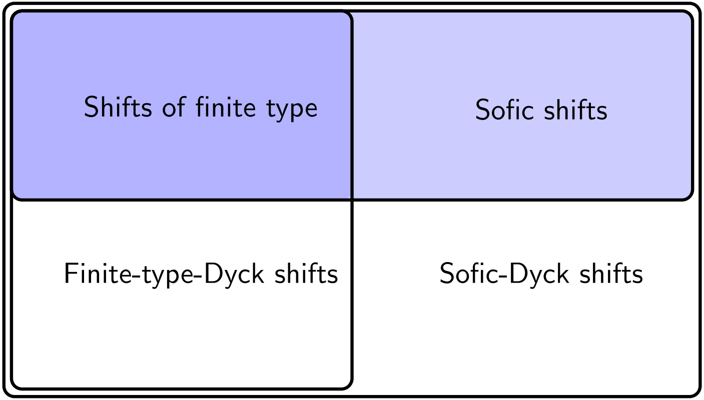
# Zeta function: summary



# Zeta function: summary



# Zeta function: summary



# Zeta function: summary

| Shifts of finite type<br>*Bowen and Lanford 1970* | Sofic shifts<br>*Manning 1971, Bowen 1978* |
|---|---|
| Dyck shift, *Keller* 1991<br>Motzkin shifts, *Inoue* 2006<br>Markov-Dyck shifts<br>*Krieger and Matsumoto* 2011 | |

# Zeta function: summary

| | |
|---|---|
| Shifts of finite type<br>*Bowen and Lanford 1970* | Sofic shifts<br>*Manning 1971, Bowen 1978* |
| Finite-type Dyck shifts | Sofic-Dyck shifts |

# Zeta function: summary

| | |
|---|---|
| Shifts of finite type<br>$\mathbb{N}$-rational | Sofic shifts<br>$\mathbb{N}$-rational, *Reutenauer 1997* |
| Finite-type Dyck shifts<br>$\mathbb{N}$-algebraic | Sofic-Dyck shifts<br>$\mathbb{N}$-algebraic |

# Topological entropy of visibly pushdown shifts

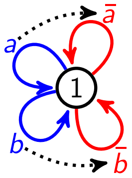
The entropy of a shift $X$ is $h(\mathcal{B}(X))$
The topological entropy of a language $L$ over $A$ is

$$h(L) = \limsup_{n \to \infty} \frac{1}{n} \log |L \cap A_n(X)|$$

Classical methods: $\mathcal{B}(X)$ is defined by a visibly pushdown grammar (hence deterministic). Well-defined $\mathbb{N}$-algebraic systems of equations allow to get $\rho$ such that $\lambda = 1/\rho$ such that $\mathcal{B}_n(X) \sim C\lambda^n n^\alpha$ and get $h(X) = \log \lambda$.

(Chomsky-Schützenberger, Kuich, Bell, Drmota, Lalley, Wood, Banderier)

# Example: the Dyck shift with 2 types of parentheses




$h(\mathcal{B}(X)) = \max(h((CA_c^*)^*), h((CA_r^*)^*), h(A_c^*), h(A_r^*))$
$C$ set of prime Dyck words

$$D = aD\bar{a}D \mid bD\bar{b}D \mid \varepsilon$$

$$C = aD\bar{a} \mid bD\bar{b}$$

$$(CA_c^*)^*(z) = \frac{2(1-2z)}{1-4z-\sqrt{1-8z^2}}$$

One gets $\rho = 1/3$ and thus $h(X) = \log 3$.

# Topological entropy of periodic patterns

The entropy of $\mathcal{P}(X)$ is $\log \frac{1}{\rho}$
where $\rho$ is the radius of convergence of $\zeta_X(z)$

for Markov-Dyck shifts
$h(X) = h(\mathcal{P}(X))$ for Markov-Dyck shifts (Krieger and Matsumoto 2011)

for visibly pushdown systems?

# Open problems and future work

It is decidable in polynomial time whether

- a sofic shift is a shift of finite type
- a regular language is strictly locally testable

A finite-type-Dyck shift is $\mathsf{X}_F$ where $F$ is a union of

- a finite set of words $G$
- a finite union of sets $u_1 c(\mathrm{Dyck}(A) \cap u_2 A^* \cap A^* v_1) r v_2$.

Is it decidable whether

- a sofic-Dyck shift is a finite-type-Dyck?
- a one-sided sofic-Dyck shift is a (one-sided) finite-type-Dyck shift?