First order and regular list functions

Laure Daviaud
University of Warwick

Joint work with Mikołaj Bojańczyk and Krishna S.

LCS seminar, Lyon, 26-04-2018

Transduction: strings to strings function

$$A^* \rightarrow B^*$$

Transduction: strings to strings function

$$A^* \to B^*$$

Two classes:

- Regular transductions
  MSOT = Det two-way transducers = copyless-SST

Transduction: strings to strings function

$$A^* \to B^*$$

Two classes:

- Regular transductions
  MSOT = Det two-way transducers = copyless-SST
- Its first-order restriction

Transduction: strings to strings function

$$A^* \to B^*$$

Two classes:

- Regular transductions
  MSOT = Det two-way transducers = copyless-SST
- Its first-order restriction

$\to$ Another characterisation of these classes.

Transduction: strings to strings function

$$A^* \to B^*$$

Two classes:

- Regular transductions
  MSOT = Det two-way transducers = copyless-SST
- Its first-order restriction

$\to$ Another characterisation of these classes.

Why ?... see later

Our running example

$$\{a, b\}^* \quad \to \quad \{a, b\}^*$$

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

number of a in w

$\overline{w}$: mirror of w

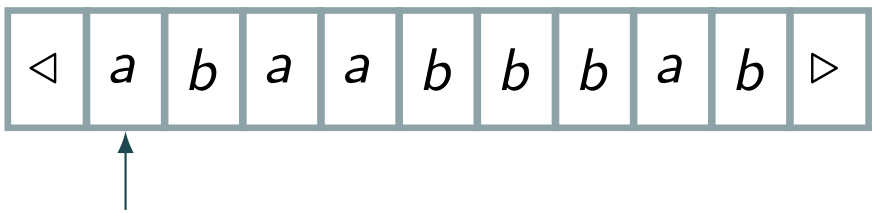
$$abaabbbab \mapsto abaabbbabbabbbaabaaaaa$$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

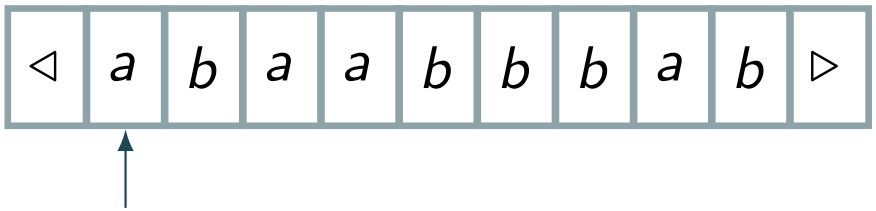
Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

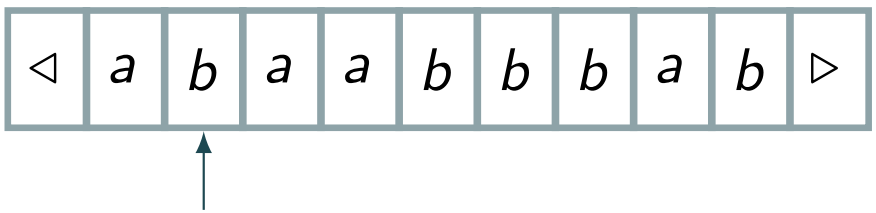
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



a

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

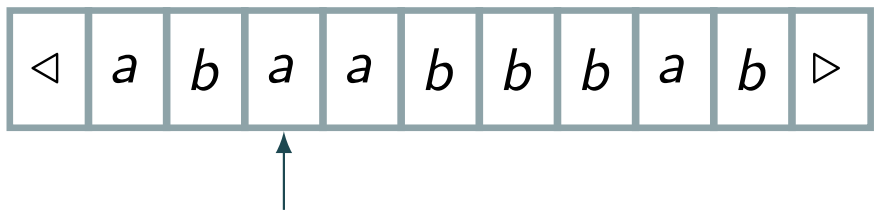
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

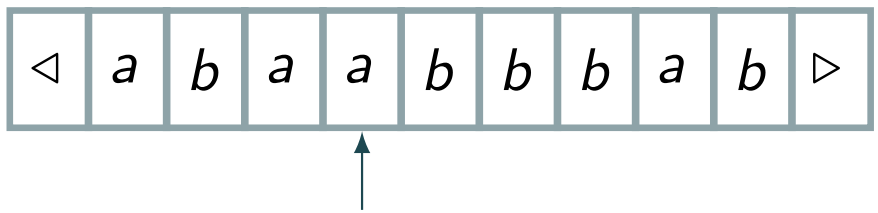
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

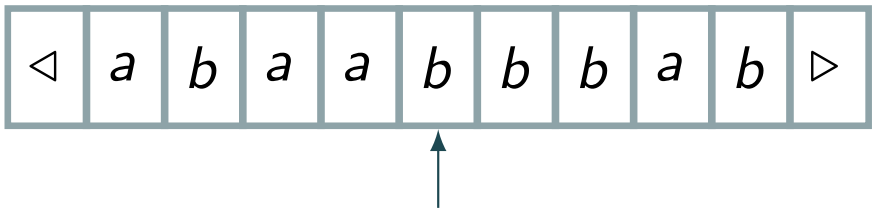
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a \quad a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

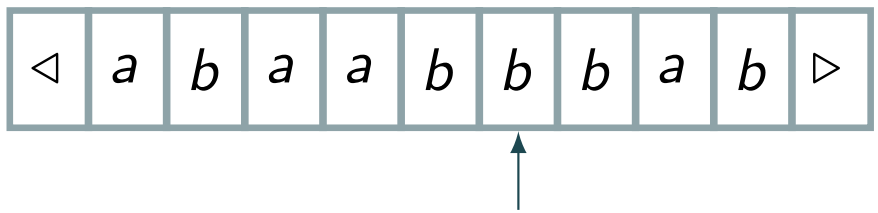
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a \quad a \quad b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

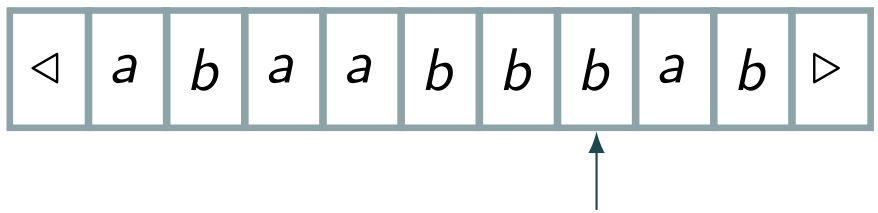
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a \quad a \quad b \quad b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

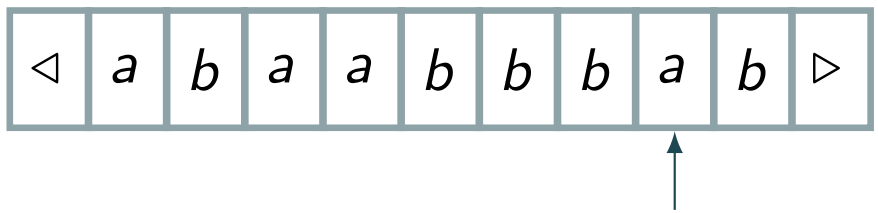
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a \quad a \quad b \quad b \quad b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

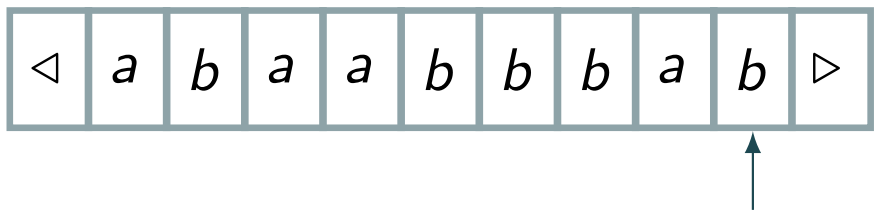
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

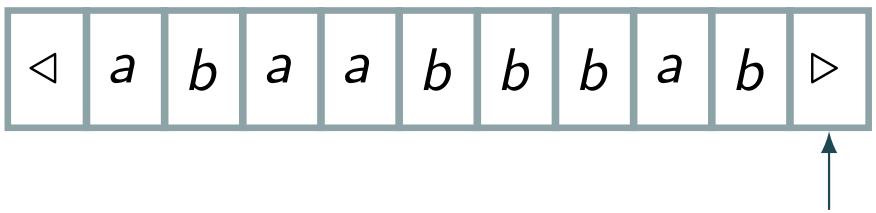
$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

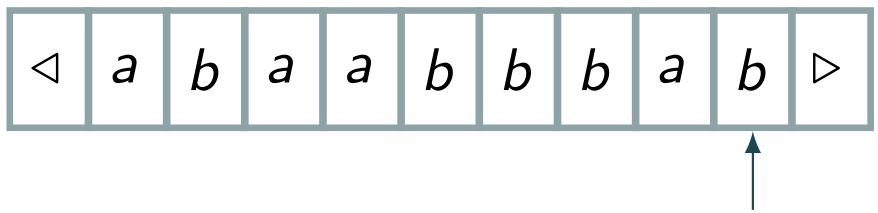
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

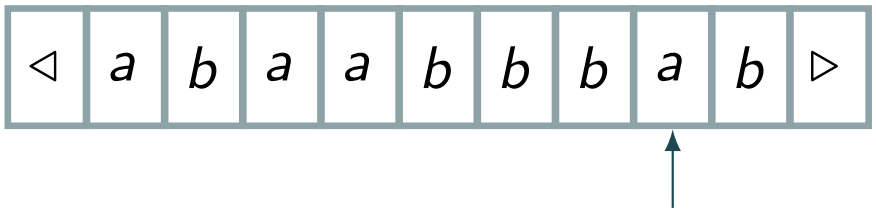
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \; b \; a \; a \; b \; b \; b \; a \; b \; b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

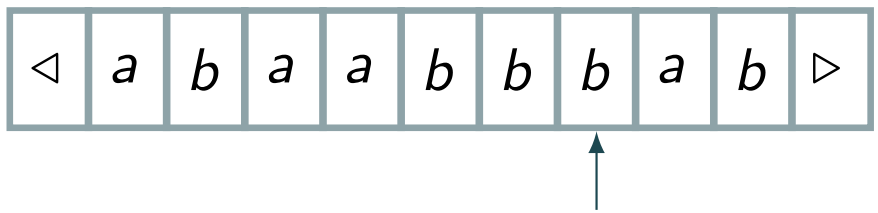
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \; b \; a \; a \; b \; b \; b \; a \; b \; b \; a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

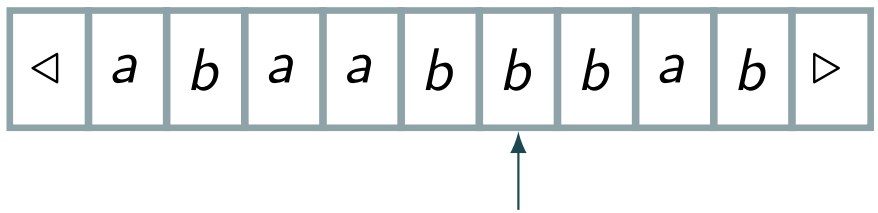
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \; b \; a \; a \; b \; b \; b \; a \; b \; b \; a \; b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

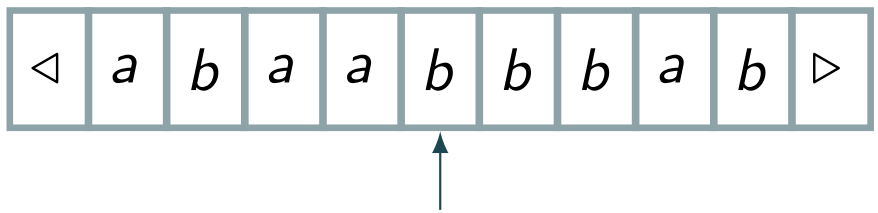
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \; b \; a \; a \; b \; b \; b \; a \; b \; b \; a \; b \; b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

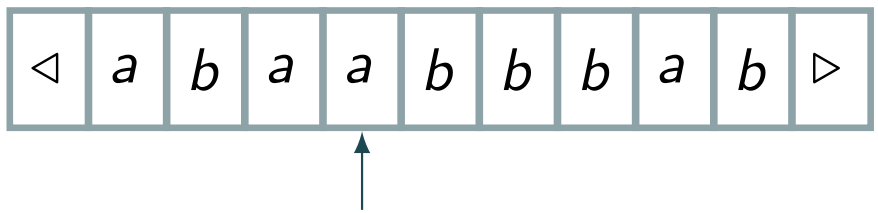
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

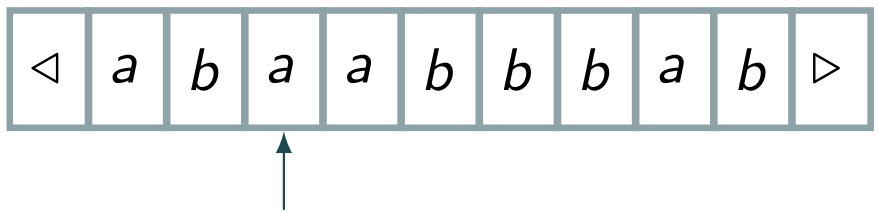
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a \; b \; a \; a \; b \; b \; b \; a \; b \; b \; a \; b \; b \; b \; a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

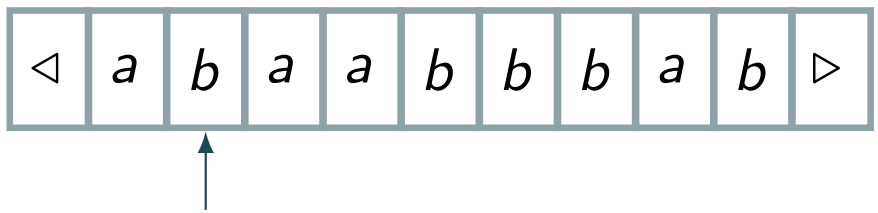
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

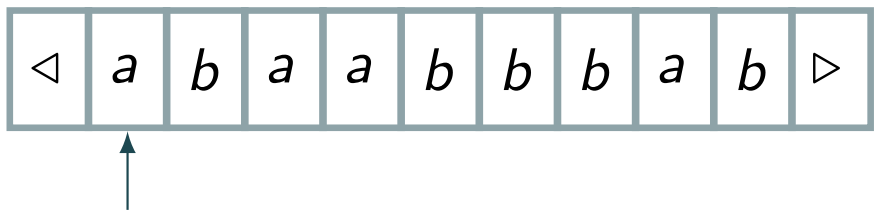
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

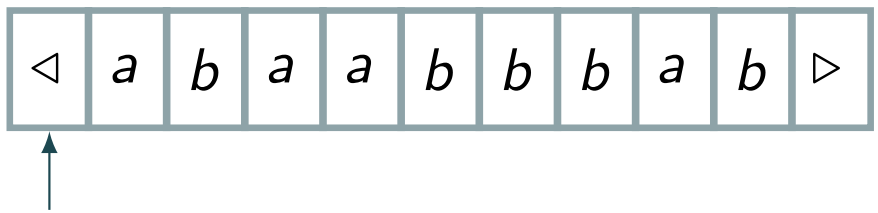
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

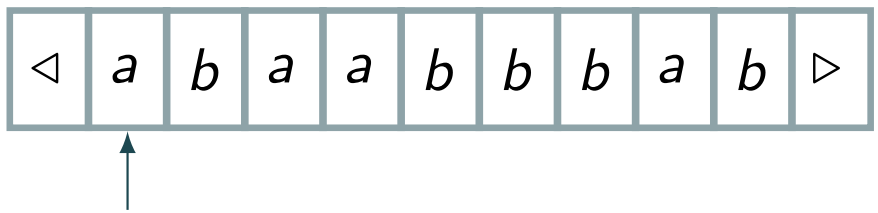
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

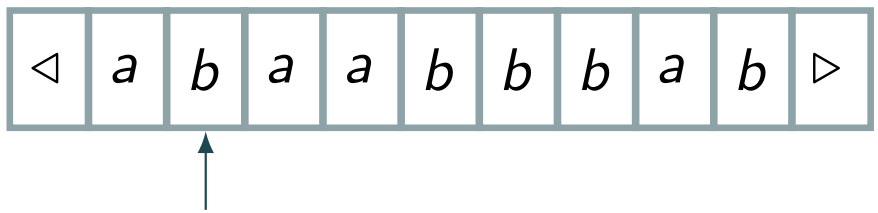
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

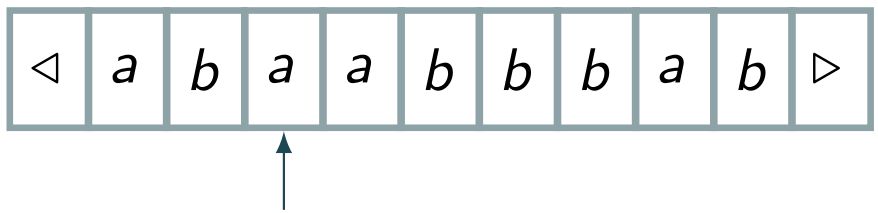
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

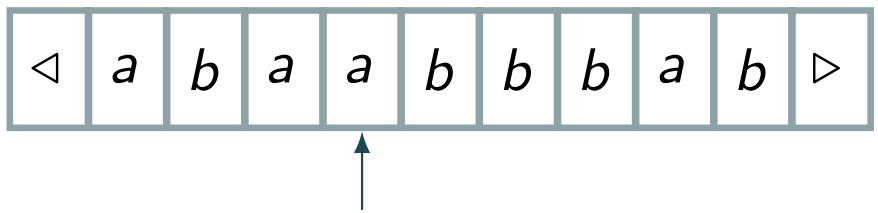
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

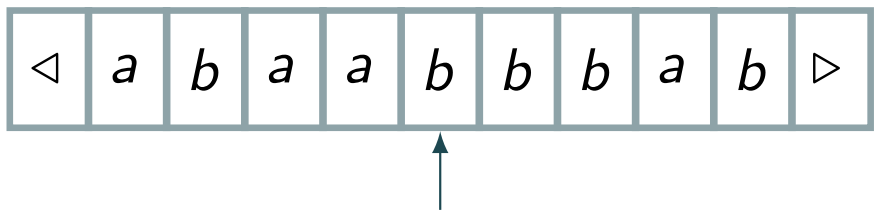
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

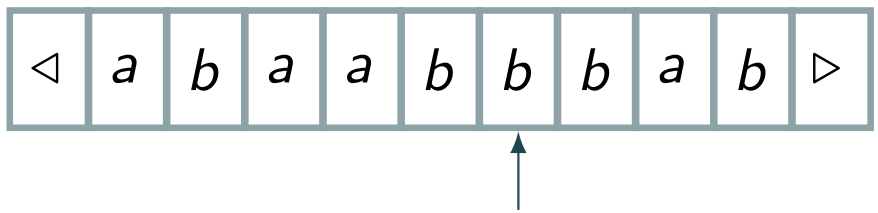
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

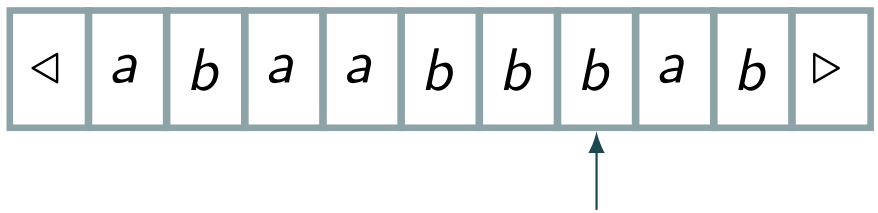
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

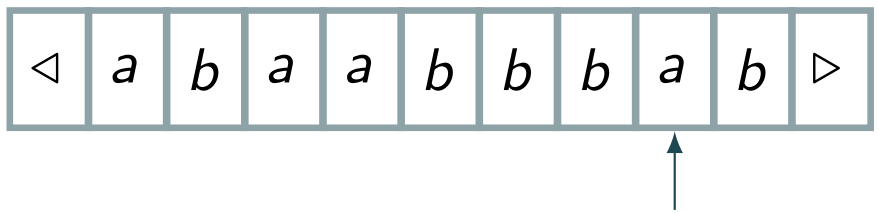
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

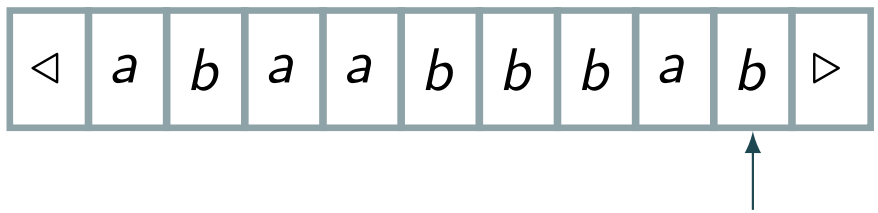
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a\ a$$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

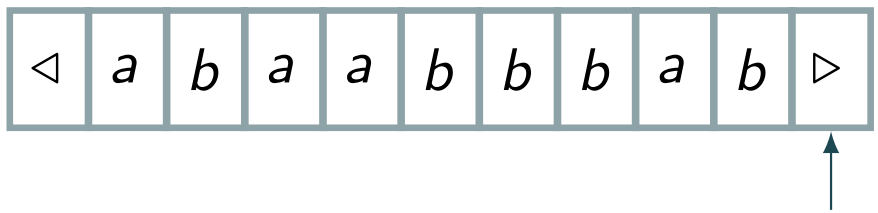
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

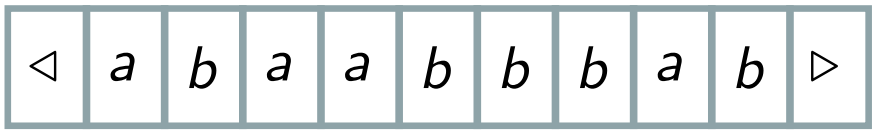
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



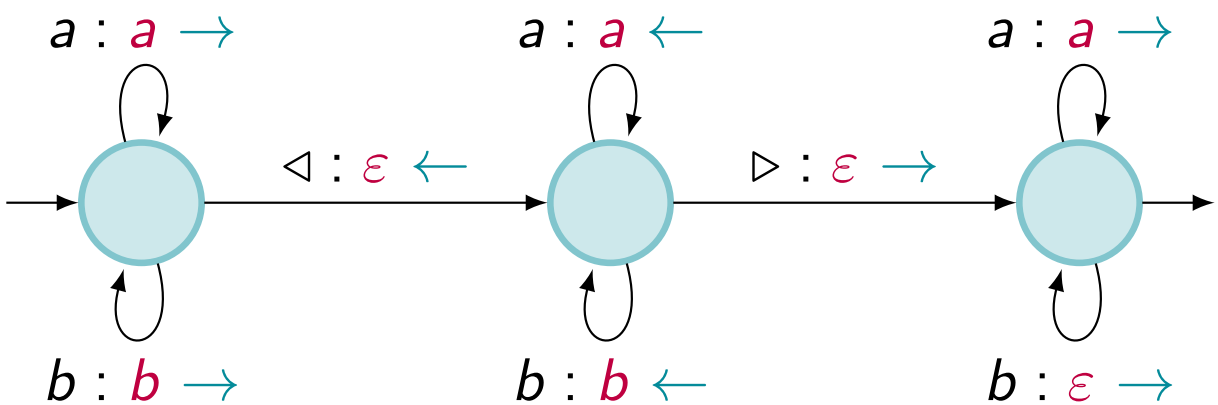
$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a\ a$

Deterministic Two-way Transducers [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$$a\ b\ a\ a\ b\ b\ b\ a\ b\ b\ a\ b\ b\ b\ a\ a\ b\ a\ a\ a\ a\ a$$

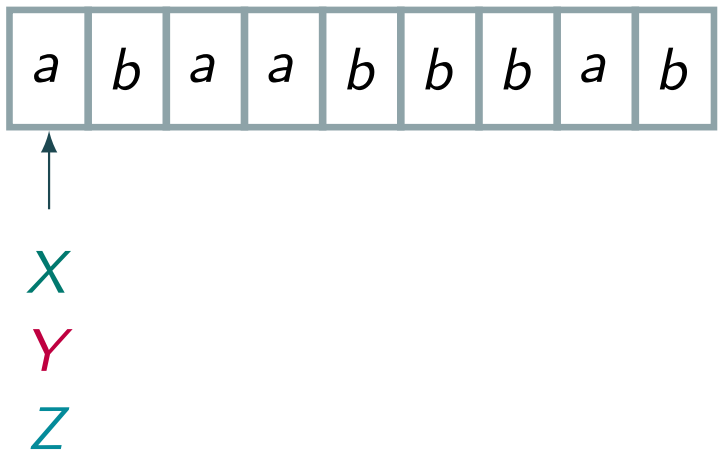


Copyless Streaming String Transducers [Alur-Cerny]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

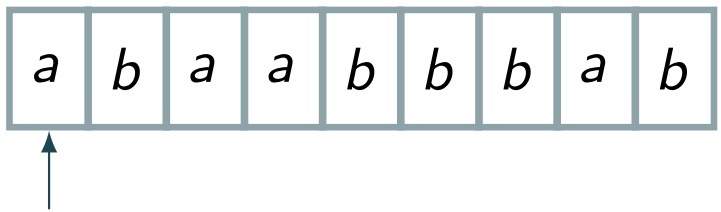
Copyless Streaming String Transducers [Alur-Cerny]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



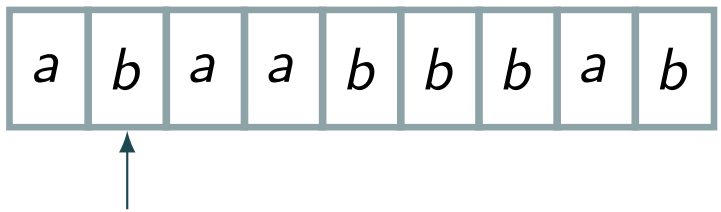
$X := a$

$Y := \qquad\qquad\qquad a$

$Z := a$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



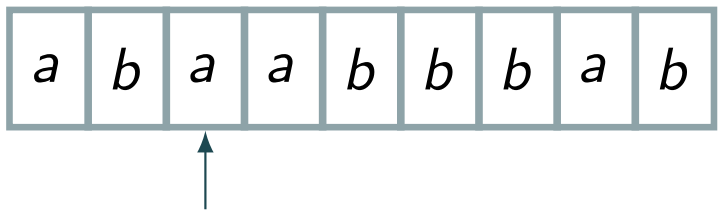
$X := a\ b$

$Y := \qquad\qquad b\ a$

$Z := a$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



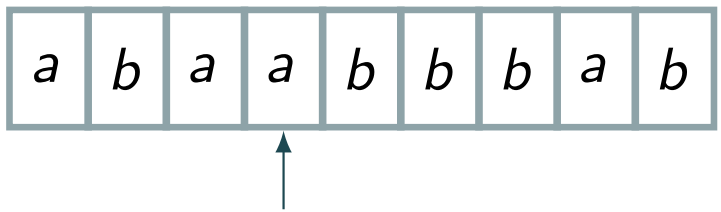
$X := a\ b\ a$

$Y := a\ b\ a$

$Z := a\ a$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



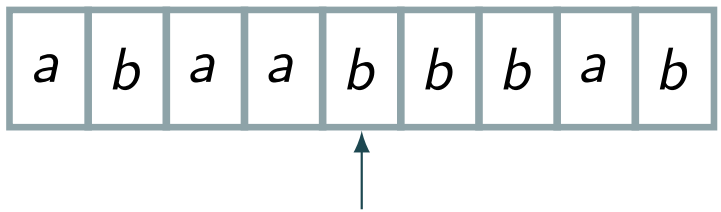
$X := a\ b\ a\ a$

$Y := \qquad\qquad a\ a\ b\ a$

$Z := a\ a\ a$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



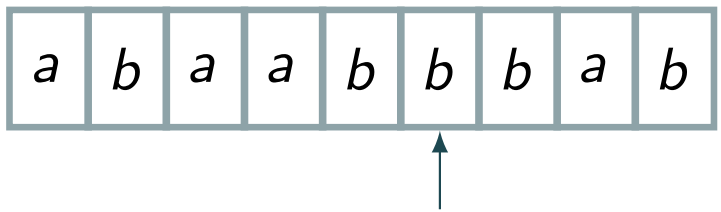
$X := a\ b\ a\ a\ b$

$Y := \qquad\qquad b\ a\ a\ b\ a$

$Z := a\ a\ a$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



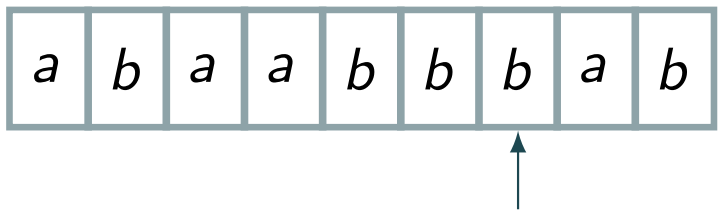
$X := a\ b\ a\ a\ b\ b$

$Y := \quad\quad\quad b\ b\ a\ a\ b\ a$

$Z := a\ a\ a$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



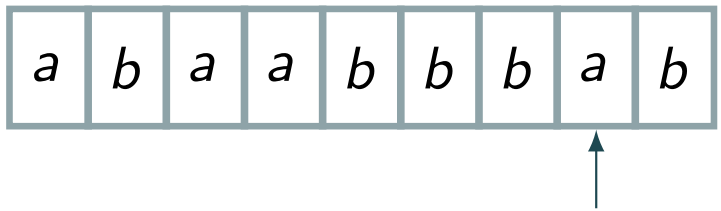
$X := a\ b\ a\ a\ b\ b\ b$

$Y := \quad\quad b\ b\ b\ a\ a\ b\ a$

$Z := a\ a\ a$

Copyless Streaming String Transducers [Alur-Cerny]

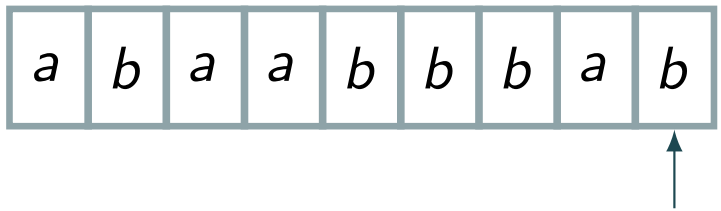
$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$



$$X := a\ b\ a\ a\ b\ b\ b\ a$$
$$Y := \quad a\ b\ b\ b\ a\ a\ b\ a$$
$$Z := a\ a\ a\ a$$

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$



$X := a\ b\ a\ a\ b\ b\ b\ a\ b$

$Y := b\ a\ b\ b\ b\ a\ a\ b\ a$

$Z := a\ a\ a\ a$

return XYZ

Copyless Streaming String Transducers [Alur-Cerny]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$

a	b	a	a	b	b	b	a	b

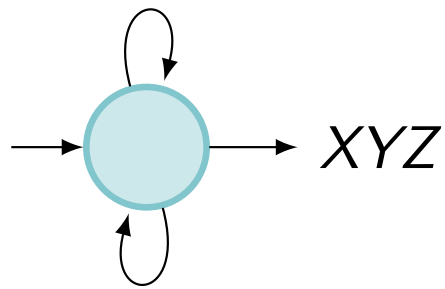
$X := a\ b\ a\ a\ b\ b\ b\ a\ b$

$Y := b\ a\ b\ b\ b\ a\ a\ b\ a$

$Z := a\ a\ a\ a$

return XYZ

$a : X := Xa \quad Y := aY \quad Z := Za$



$b : X := Xb \quad Y := bY \quad Z := Z$

Copyless:

$\begin{cases} X := XY \\ Y := ZX \end{cases}$

MSO-transductions [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

logical structure on words
$<, P_a, P_b$

$$a \longrightarrow b \longrightarrow a \longrightarrow a \longrightarrow b \longrightarrow b \longrightarrow b \longrightarrow a \longrightarrow b$$

MSO-transductions [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

- Make k copies of the input

MSO-transductions [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

- Make k copies of the input
- Select positions in the output with MSO-formulas

MSO-transductions [Engelfriet-Hoogeboom]

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

- Make k copies of the input
- Select positions in the output with MSO-formulas
- Define an output word using MSO-formulas

MSO-transductions [Engelfriet-Hoogeboom]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

$$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$$

- Make k copies of the input
- Select positions in the output with MSO-formulas
- Define an output word using MSO-formulas

MSO-transductions [Engelfriet-Hoogeboom]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$

$a \to b \to a \to a \to b \to b \to b \to a \to b$

$a \to b \to a \to a \to b \to b \to b \to a \to b$

$a \leftarrow b \leftarrow a \leftarrow a \leftarrow b \leftarrow b \leftarrow b \leftarrow a \leftarrow b$

$a \quad b \quad a \quad a \quad b \quad b \quad b \quad a \quad b$

- Make k copies of the input
- Select positions in the output with MSO-formulas
- Define an output word using MSO-formulas

MSO-transductions [Engelfriet-Hoogeboom]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \leftarrow b \leftarrow a \leftarrow a \leftarrow b \leftarrow b \leftarrow b \leftarrow a \leftarrow b$$

$$a \longrightarrow b \longrightarrow a \to a \longrightarrow b \longrightarrow b \longrightarrow b \longrightarrow a \quad b$$

- Make k copies of the input
- Select positions in the output with MSO-formulas
- Define an output word using MSO-formulas

MSO-transductions [Engelfriet-Hoogeboom]

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$
$$\downarrow$$
$$a \leftarrow b \leftarrow a \leftarrow a \leftarrow b \leftarrow b \leftarrow b \leftarrow a \leftarrow b$$
$$\downarrow$$
$$a \longrightarrow b \longrightarrow a \to a \longrightarrow b \longrightarrow b \longrightarrow b \longrightarrow a \quad b$$

- Make k copies of the input
- Select positions in the output with MSO-formulas
- Define an output word using MSO-formulas

MSO-transductions [Engelfriet-Hoogeboom]

FO

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

$$a \to b \to a \to a \to b \to b \to b \to a \to b \downarrow$$

$$a \leftarrow b \leftarrow a \leftarrow a \leftarrow b \leftarrow b \leftarrow b \leftarrow a \leftarrow b \downarrow$$

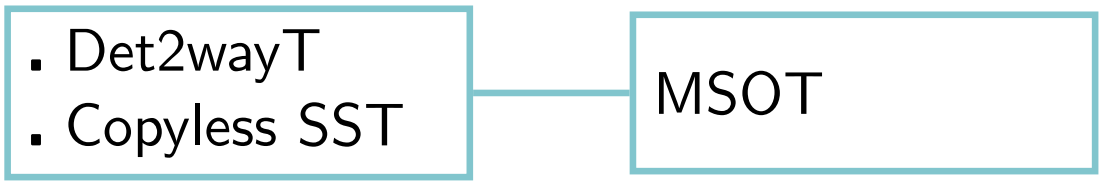
$$a \to b \to a \to a \to b \to b \to b \to a \to b$$

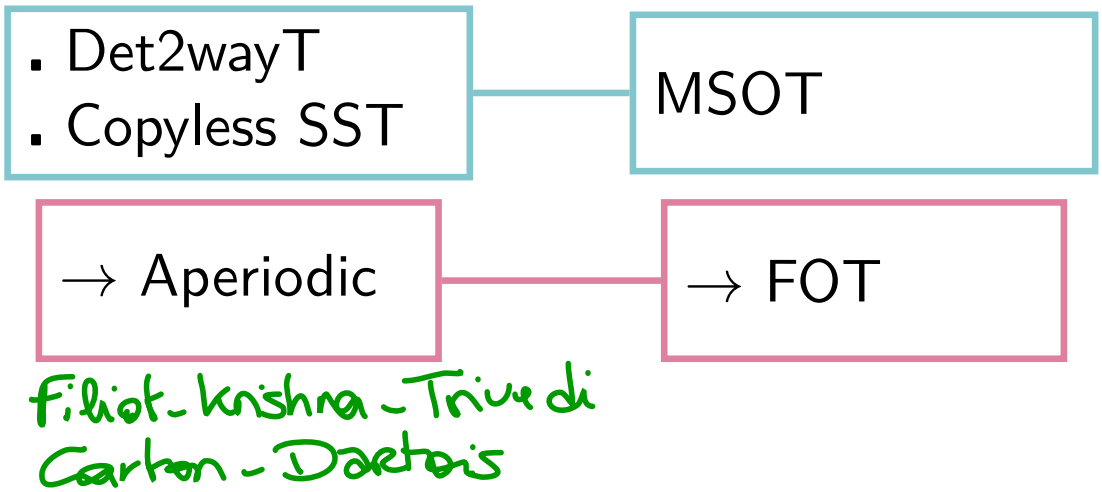
- Make k copies of the input
- Select positions in the output with MSO-formulas
- Define an output word using MSO-formulas

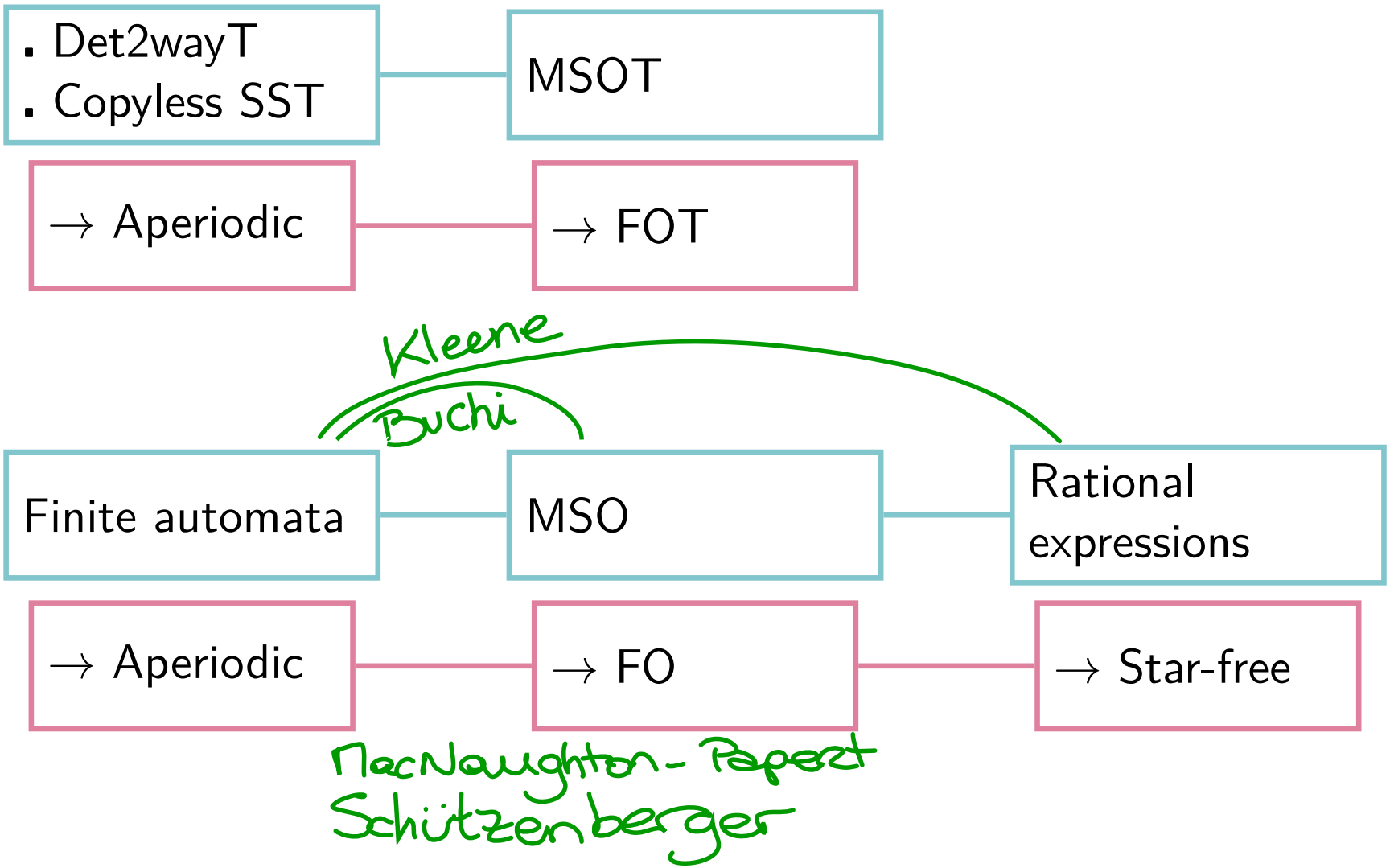
FO

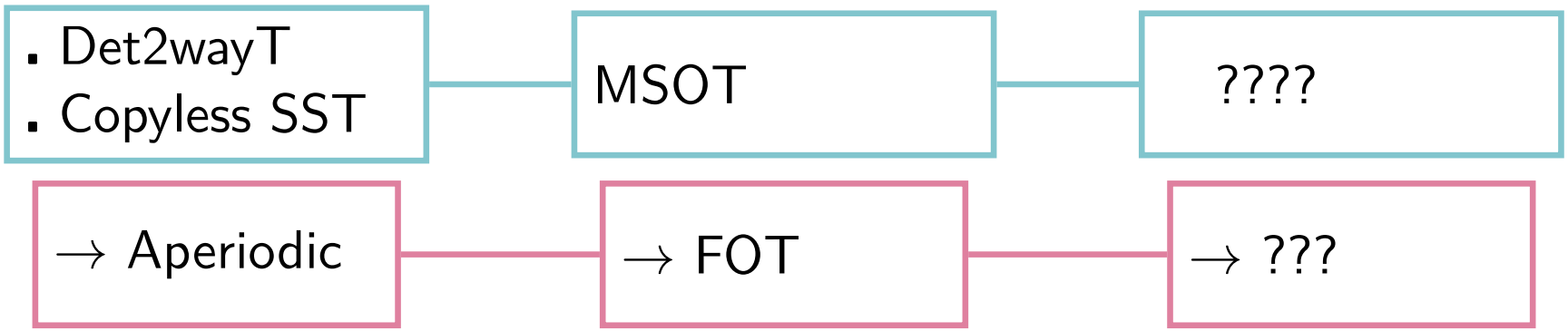
FO

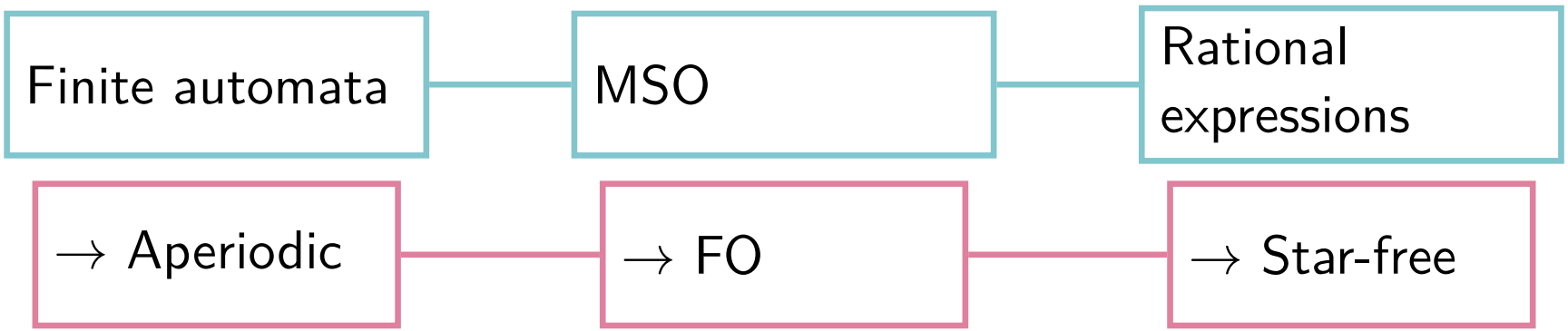
Is the example FO?


Det2wayT
Copyless SST
MSOT


. Det2wayT
. Copyless SST
MSOT
→ Aperiodic
→ FOT
Filiot-Krishna-Trivedi
Carton-Dartois


. Det2wayT
. Copyless SST
MSOT
→ Aperiodic
→ FOT
Kleene
Buchi
Finite automata
MSO
Rational expressions
→ Aperiodic
→ FO
→ Star-free
MacNaughton-Papert
Schützenberger


. Det2wayT
. Copyless SST
MSOT
????
→ Aperiodic
→ FOT
→ ???


Finite automata
MSO
Rational expressions
→ Aperiodic
→ FO
→ Star-free

Functions on a type system

$\longrightarrow$ Functions on lists, lists of lists, pairs of lists...

Functions on a type system

$\longrightarrow$ Functions on lists, lists of lists, pairs of lists...

$$\mathcal{T} := \text{every one-element set} \quad | \quad \mathcal{T} + \mathcal{T} \quad | \quad \mathcal{T} \times \mathcal{T} \quad | \quad \mathcal{T}^*$$

Functions on a type system

$\longrightarrow$ Functions on lists, lists of lists, pairs of lists...

$$\mathcal{T} := \text{every one-element set} \quad | \quad \mathcal{T} + \mathcal{T} \quad | \quad \mathcal{T} \times \mathcal{T} \quad | \quad \mathcal{T}^*$$

Basic functions:

• **projection**$_1$	• **coprojection**	• **distribute**
$\Sigma \times \Gamma \longrightarrow \Sigma$	$\Sigma \longrightarrow \Sigma + \Gamma$	$(\Sigma + \Gamma) \times \Delta \longrightarrow (\Sigma \times \Delta) + (\Gamma \times \Delta)$
$(x, y) \longmapsto x$	$x \longmapsto x$	$(x, y) \longmapsto (x, y)$

Basic functions on lists

Reverse. $[a, b, a, a, b, c] \mapsto [c, b, a, a, b, a]$

Flat. $[[a, b], [c]] \mapsto [a, b, c]$

Append. $(a, [b, b, c, a]) \mapsto [a, b, b, c, a]$

Co-append. $[a, b, b, c, a] \mapsto (a, [b, b, c, a])$

Block. $[a, b, c, d, a, a, b, c, d] \mapsto [[a, b], [c, d], [a, a, b], [c, d]]$

Operations

Disjoint union.

$$\frac{f : \Sigma \longrightarrow \Delta \qquad g : \Gamma \longrightarrow \Delta}{f + g : \Sigma + \Gamma \longrightarrow \Delta} \qquad x \mapsto \begin{cases} f(x) & \text{if } x \in \Sigma \\ g(x) & \text{if } x \in \Gamma \end{cases}$$

Composition.

$$\frac{f : \Sigma \longrightarrow \Gamma \qquad g : \Gamma \longrightarrow \Delta}{g \circ f : \Sigma \longrightarrow \Delta} \qquad x \mapsto g(f(x))$$

Map.

$$\frac{f : \Sigma \longrightarrow \Gamma}{f^* : \Sigma^* \longrightarrow \Gamma^*} \qquad [x_1, \ldots, x_n] \mapsto \begin{cases} [f(x_1), \ldots, f(x_n)] & \text{if } n > 0 \\ [] & \text{if } n = 0 \end{cases}$$

Pairing.

$$\frac{f : \Sigma \longrightarrow \Gamma \qquad g : \Sigma \longrightarrow \Delta}{(f, g) : \Sigma \longrightarrow \Gamma \times \Delta} \qquad x \mapsto (f(x), g(x))$$

First-order list functions

First-order list functions:
Smallest class of functions over any Σ from $\mathcal{T}$, which contains:

- all the constant functions,
- the functions projection, co-projection and distribute,
- the functions reverse, flat, append, co-append and block,

and which is closed under applying the operations:

- disjoint union,
- composition,
- map,
- pairing.

First-order list functions

First-order list functions:
Smallest class of functions over any Σ from $\mathcal{T}$, which contains:

- all the constant functions,
- the functions projection, co-projection and distribute,
- the functions reverse, flat, append, co-append and block,

and which is closed under applying the operations:

- disjoint union,
- composition,
- map,
- pairing.

Theorem

The first-order list functions are exactly the FO-transductions.

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$
- $[a, b, a, a, b, b, b, a, b] \mapsto [a, a, a, a]$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$
- $[a, b, a, a, b, b, b, a, b] \mapsto [a, a, a, a]$
  Block: $[a, b, a, a, b, b, b, a, b] \mapsto [[a], [b], [a, a], [b, b, b], [a], [b]]$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$
- $[a, b, a, a, b, b, b, a, b] \mapsto [a, a, a, a]$
  Block: $[a, b, a, a, b, b, b, a, b] \mapsto [[a], [b], [a, a], [b, b, b], [a], [b]]$
  Disjoint union of the identity of a^* and $b^* \mapsto [] \in a^*$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$
- $[a, b, a, a, b, b, b, a, b] \mapsto [a, a, a, a]$
  Block: $[a, b, a, a, b, b, b, a, b] \mapsto [[a], [b], [a, a], [b, b, b], [a], [b]]$
  Disjoint union of the identity of a^* and $b^* \mapsto [] \in a^*$
  Map: $[[a], [b], [a, a], [b, b, b], [a], [b]] \mapsto [[a], [], [a, a], [], [a], []]$

Let's go back to our running example

$$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$
- $[a, b, a, a, b, b, b, a, b] \mapsto [a, a, a, a]$
  Block: $[a, b, a, a, b, b, b, a, b] \mapsto [[a], [b], [a, a], [b, b, b], [a], [b]]$
  Disjoint union of the identity of a^* and $b^* \mapsto [] \in a^*$
  Map: $[[a], [b], [a, a], [b, b, b], [a], [b]] \mapsto [[a], [], [a, a], [], [a], []]$
  Flat: $[[a], [], [a, a], [], [a], []] \mapsto [a, a, a, a]$

Let's go back to our running example

$w \quad \mapsto \quad w\overline{w}a^{|w|_a}$

- Identity: $[a, b, a, a, b, b, b, a, b] \mapsto [a, b, a, a, b, b, b, a, b]$
- Reverse: $[a, b, a, a, b, b, b, a, b] \mapsto [b, a, b, b, b, a, a, b, a]$
- $[a, b, a, a, b, b, b, a, b] \mapsto [a, a, a, a]$
  Block: $[a, b, a, a, b, b, b, a, b] \mapsto [[a], [b], [a, a], [b, b, b], [a], [b]]$
  Disjoint union of the identity of a^* and $b^* \mapsto [] \in a^*$
  Map: $[[a], [b], [a, a], [b, b, b], [a], [b]] \mapsto [[a], [], [a, a], [], [a], []]$
  Flat: $[[a], [], [a, a], [], [a], []] \mapsto [a, a, a, a]$

$\rightarrow$ Pairing, transform into lists and flat

Extension to MSO

Regular list functions:
Same as the first-order list function, except that for every group, we add its prefix multiplication function to the base functions.

Extension to MSO

Regular list functions:
Same as the first-order list function, except that for every group, we add its prefix multiplication function to the base functions.

Prefix multiplication operation:
$[g_1, g_2, g_3, \ldots, g_n] \mapsto [g_1, g_1g_2, g_1g_2g_3, \ldots, g_1g_2 \cdots g_n]$

Extension to MSO

Regular list functions:
Same as the first-order list function, except that for every group, we add its prefix multiplication function to the base functions.

Prefix multiplication operation:
$[g_1, g_2, g_3, \ldots, g_n] \mapsto [g_1, g_1g_2, g_1g_2g_3, \ldots, g_1g_2 \cdots g_n]$

Theorem

The regular list functions are exactly the MSO-transductions.

Conclusion

Conclusion

Benefit of this new characterisation:

Conclusion

Benefit of this new characterisation:

- simpler and more natural choice of basic functions and combinators (thanks to the type system)

Conclusion

Benefit of this new characterisation:

- simpler and more natural choice of basic functions and combinators (thanks to the type system)
- link with functional programming languages

Conclusion

Benefit of this new characterisation:

- simpler and more natural choice of basic functions and combinators (thanks to the type system)
- link with functional programming languages
- Identify the iteration mechanism

Conclusion

Benefit of this new characterisation:

- simpler and more natural choice of basic functions and combinators (thanks to the type system)
- link with functional programming languages
- Identify the iteration mechanism

Extensions:

- Other types, $\mathbb{N}$...?
- Non-linear operators?
- Other data structures, trees...?