Word transducers: from 2-way to 1-way

Anca Muscholl

joint work with F. Baschenis, O. Gauwin, G. Puppis (Bordeaux),
to appear at FSTTCS'15

Transductions

transform objects - here: words

transduction: mapping (or relation) from words to words

dagstuhl	⟶	dgsthl	erase vowels
dagstuhl	⟶	lhutsgad	reverse
dagstuhl	⟶	dagstuhldagstuhl	duplicate
dagstuhl	⟶	stuhldag	permute circularly

Transducers

FST: finite state transducers, one-way

dagstuhl	→	dgsthl	erase vowels

2GSM: two-way transducers (_g_eneralized _s_equential _m_appings)

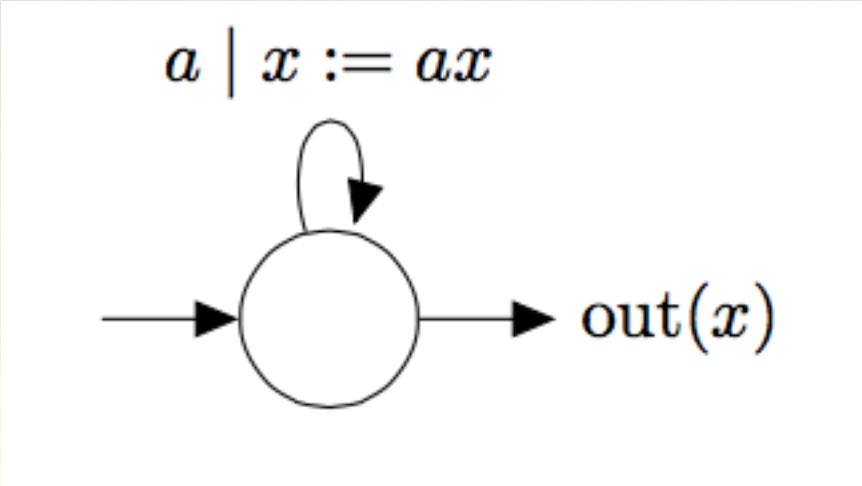
dagstuhl	→	lhutsgad	reverse
dagstuhl	→	dagstuhldagstuhl	duplicate

Transducers

SST: streaming transducers (AlurCerny 2010)

- one-way
- use a finite number of (copyless) registers, where output can be appended left or right

reverse



dagstuhl ⟶ lhutsgad

Transductions

MSOT: monadic second-order transductions (Courcelle)

maps structures into structures (bounded copy)

- domain formula: unary MSO formula "c-th copy of input position belongs to the output and is labeled by a"
- order formula: binary MSO formula "c-th copy of position x precedes the d-th copy of position y in the output"

Transductions

MSOT: monadic second-order transductions (Courcelle)

Ex: reverse transformation

- domain formula: $\mathrm{dom}_a(x) \equiv a(x)$
- order formula: $\mathrm{Less}(x, y) \equiv (x > y)$

Automata = logic

- [Engelfriet, Hoogeboom 2001]

2DGSM = MSOT

- [Alur, Cerny 2010]

DSST = MSOT

- Non-determinism

NSST = NMSOT, incomparable with 2NGSM

but: NSST = DSST for functional transducers

First-order transductions (FOT)

- [McKenzie, Schwentick, Thérien, Vollmer 2006]

  aperiodic NFT = FO translations

- [Filiot, Shankara Narayanan, Trivedi 2014]

  SST with aperiodic transition monoid = FOT

- [Carton, Dartois 2014]

  aperiodic 2DGSM = FOT

Transducers and resources

Resources for 2GSM and SST:

- 2GSM: maximal number of visits on a position
- SST: number of registers

[Ledent, M., Salvati 2013]

DSST with k registers = 2DGSM with visit number (2k+1)

(DSST with regular look-ahead, 2DGSM with regular look-around)

Resources: passes over input

problem: given a 2NGSM, is it equivalent to some NFT?

The above problem is decidable for functional 2NGSM, with non-elementary complexity.

[Filiot, Gauwin, Reynier, Servais 2013]

2-way vs. 1-way

We show: given a sweeping★ 2NGSM

- 2EXPSPACE algorithm deciding if an equivalent NFT exists
- if "yes", produce an equivalent NFT of 2-exp. size

[Baschenis, Gauwin, M., Puppis 2015]

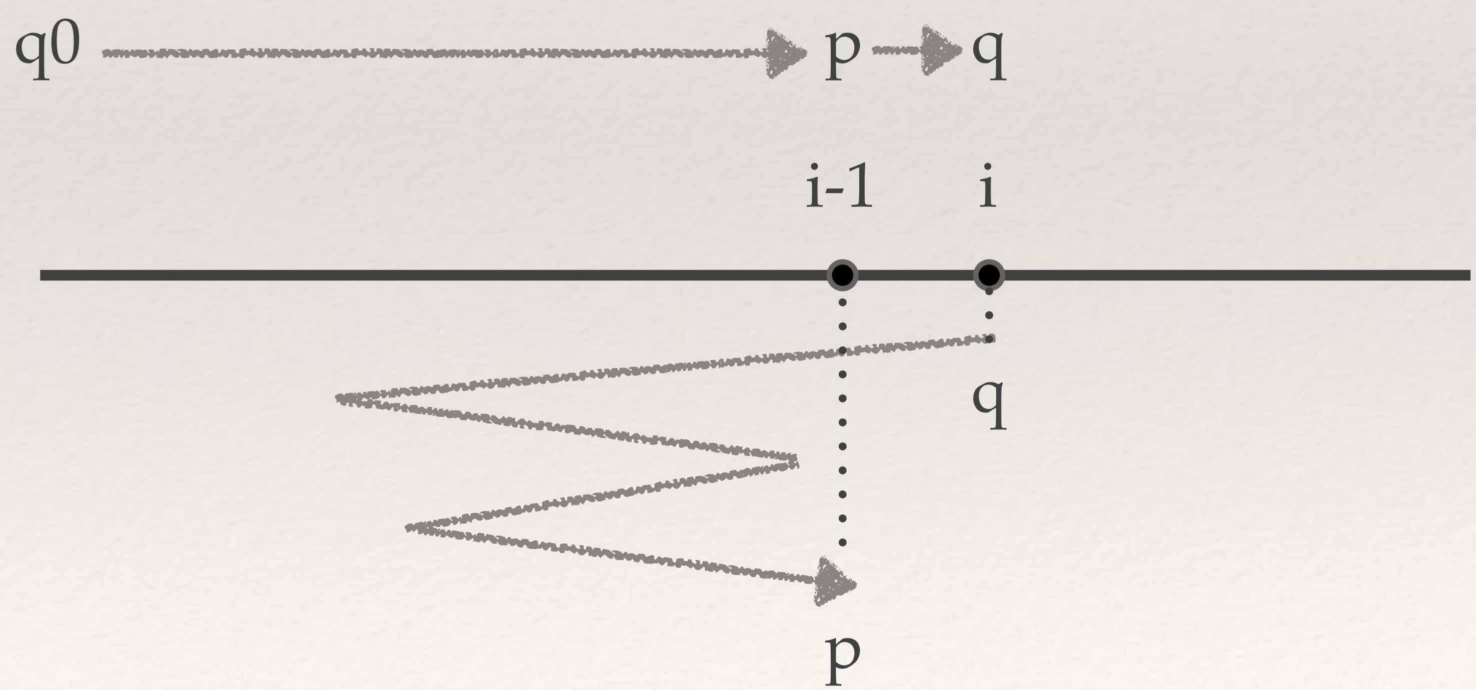
★ sweeping: U-turns only at borders

Example: 2-way

Fix a regular language R.

F(u v w) = mirror(v), v = the rightmost maximal factor in R

2DGSM



[Hopcroft,Ullman '67]

Example: 1-way

Fix a regular language R.

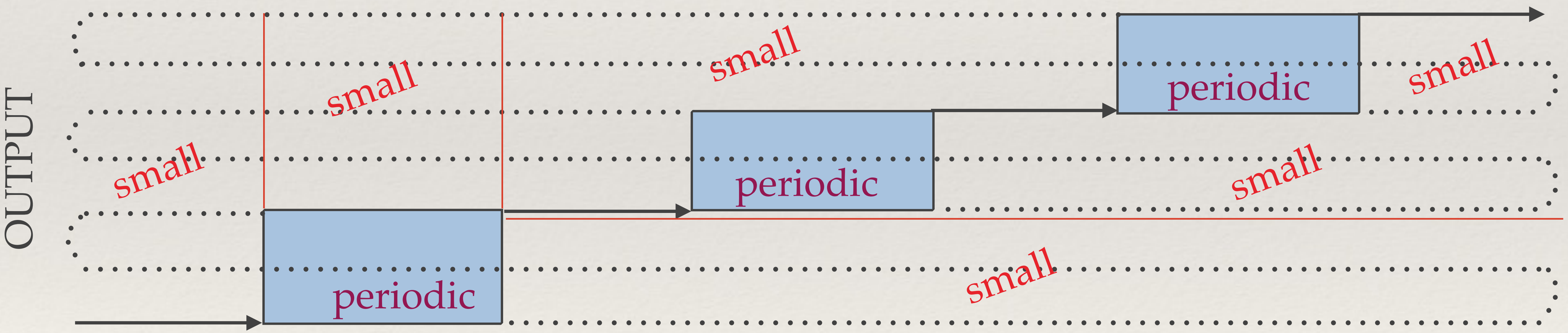
F(u v w) = mirror(v), v = the rightmost maximal factor in R

There is an NFT for F if R = (ab)*:

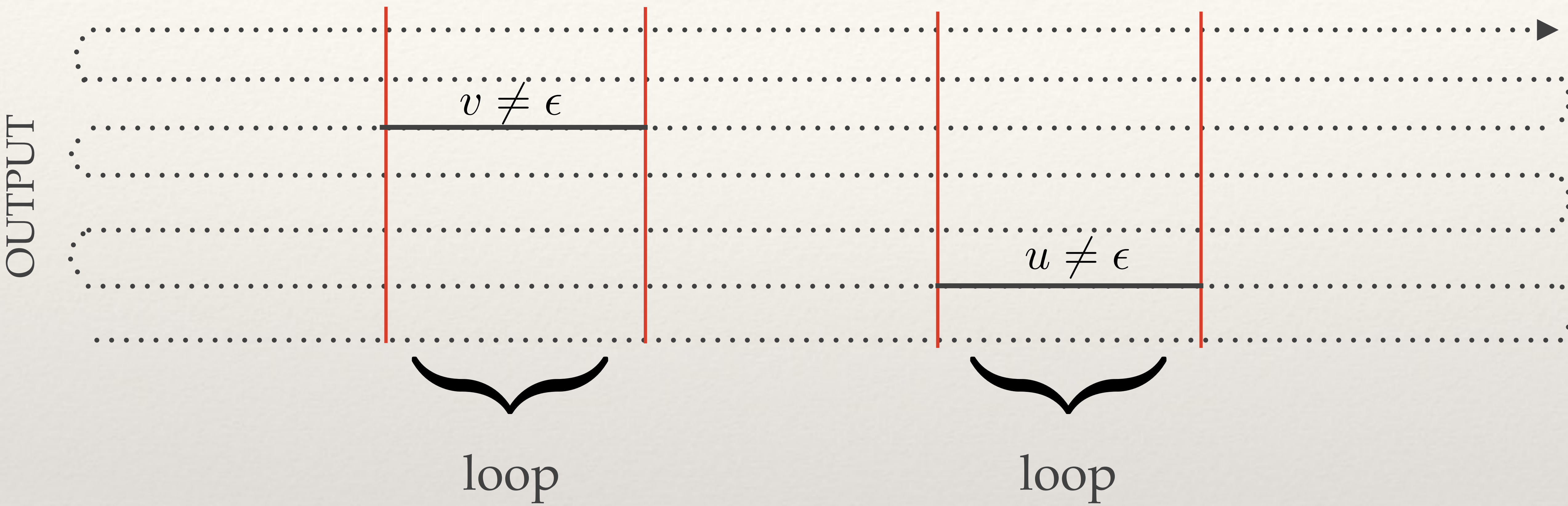
- guess the beginning of v
- output (ba)* until the end of v
- check w

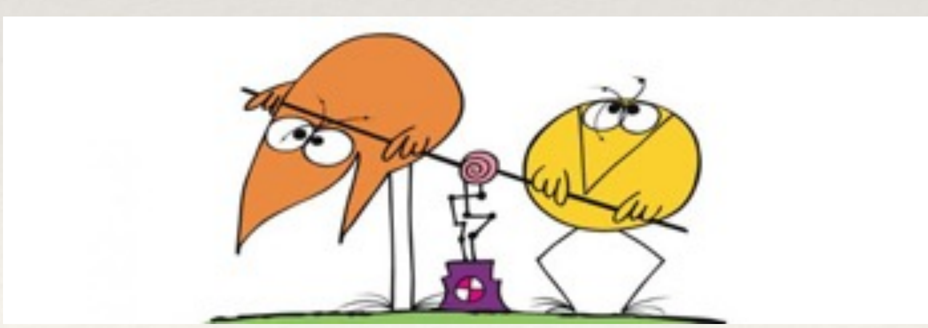
2-way vs. 1-way

A 2NGSM is equivalent to some NFT if every accepting run has a decomposition into left-to-right passes and blocks with periodic output.



Inversions





$$x\ u^{kM+c}\ y\ v^{kN+d}\ w = \alpha\ \beta^{N}\ \gamma\ \delta^{M}\ \zeta$$

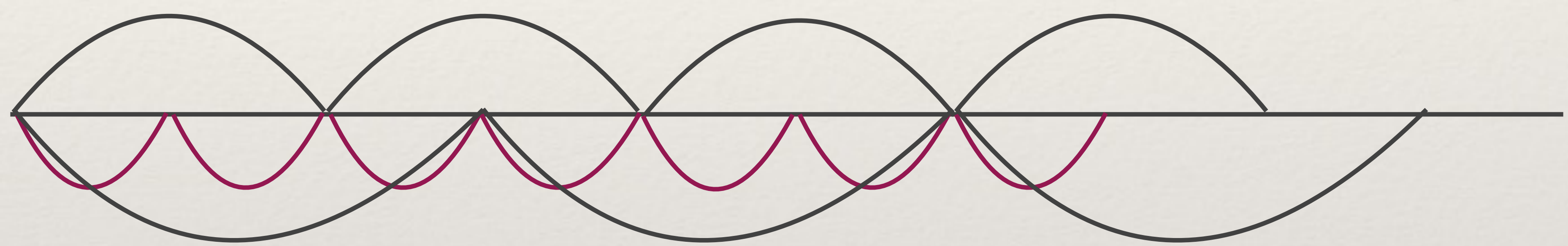
sweeping transducer

1-way transducer

Inversions

$$x\ u^{kM+c}\ y\ v^{kN+d}\ w = \alpha\ \beta^{N}\ \gamma\ \delta^{M}\ \zeta$$

Fine and Wilf



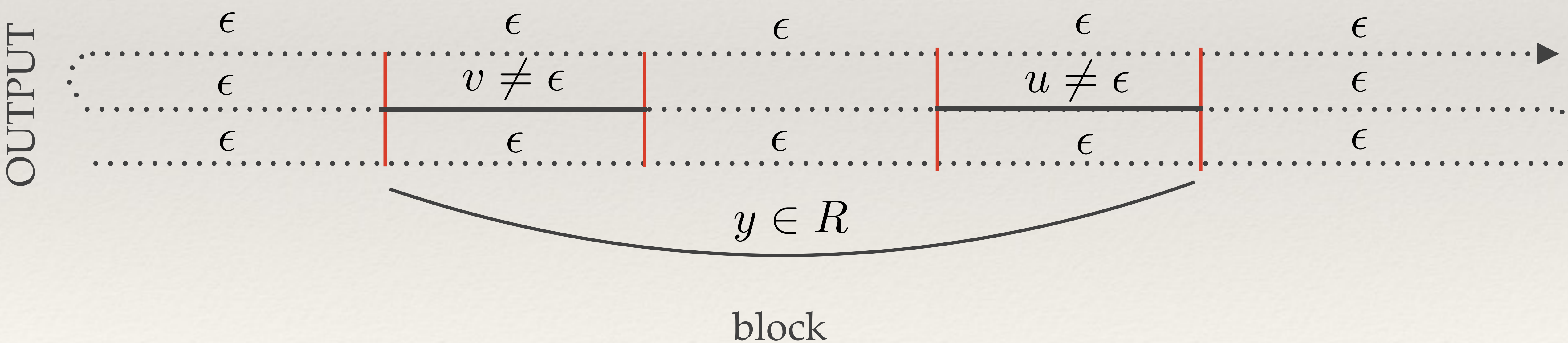
If a very long word has periods p and q, then also the period gcd(p,q).

Some applications of Fine&Wilf show that the output uyv of an inversion is periodic (with period bounded by the sweeping transducer). Two overlapping inversions will have an overall periodic output, which leads to the decomposition into blocks.

Example

Regular language $R \subseteq (ab)^*$

F(x y z) = mirror(v), y = the rightmost maximal factor in R



Characterization

Thm. It can be checked in EXPSPACE if a sweeping 2NGSM A is equivalent to some NFT. An equivalent, 2-exp. size NFT B can be constructed.

Algorithm: construct B and check if dom(A) = dom(B).

Optimal:

$$F(\, a_0 \;\text{bin}(0)\; a_1 \;\text{bin}(1) \ldots\; a_{2^N-1} \;\text{bin}(\, 2^N - 1\,)) = \; a_0 \cdots a_{2^n-1} a_0 \cdots a_{2^n-1}$$

Sweeping vs. streaming

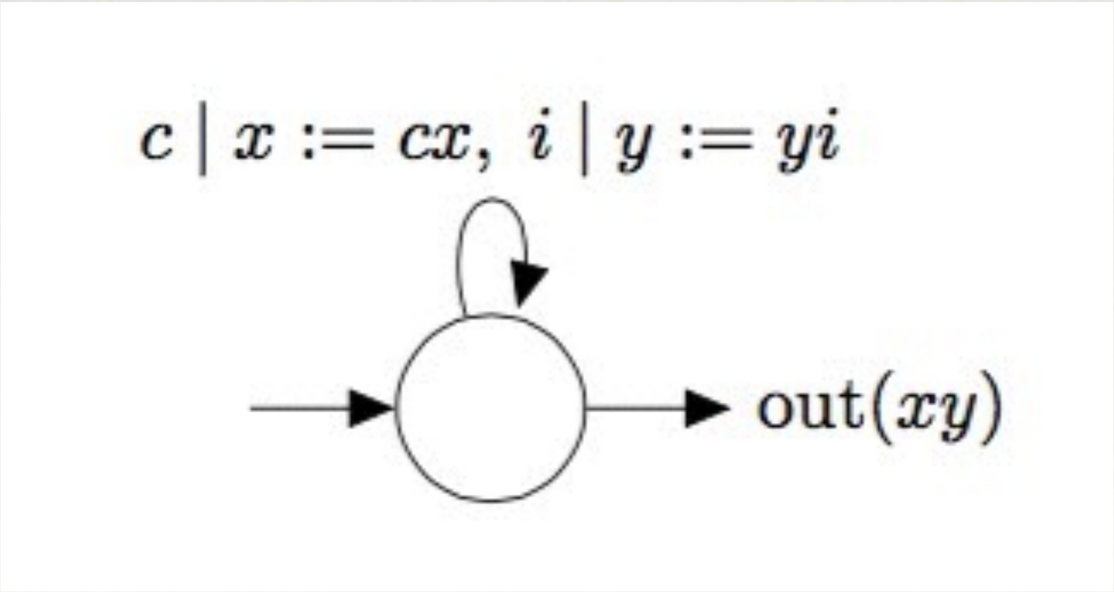
SST: streaming transducers

Example:

$$u \in \{a, b, 0, 1\}^*$$

F(u) = mirror(v) w

$$v = \pi_{a,b}(u) \qquad w = \pi_{0,1}(u)$$



$$c \in \{a, b\}, i \in \{0, 1\}$$

This SST is non-mixing: no variable concatenation in the updates.

Sweeping and streaming

For any word-to-word function F tfae:

1. F can be computed by a non-mixing, unambiguous streaming transducer
2. F can be computed by an unambiguous sweeping transducer

1. k registers ⟶ 2k+1 passes
2. k passes ⟶ k registers

Characterization

Thm. It can be checked in EXPSPACE if a non-mixing, unambiguous SST A is equivalent to some NFT. An equivalent, 2-exp. size NFT B can be constructed.

Algorithm: construct an equivalent sweeping transducer A′ (size of A′ is polynomial in size of A). Apply the algorithm for sweeping transducers.

Ongoing work

- Problem 1: given a sweeping transducer, compute the minimal number of passes.
- Problem 2: given an SST, compute the minimal number of variables.

1: decide the existence of j-block decompositions

2: Transform the SST into a sweeping transducer and apply 1. Result is at most twice the minimal number.

Conclusion

- We showed an elementary construction from 2-way to 1-way.
- Conjecture: similar decision procedure for arbitrary 2NGSM (non-sweeping). But the combinatorics gets very complicated (pumping!).
- In the "origin semantics" (Bojanczyk) the problem is easier: PSPACE.
- Open: characterizing first-order definable transductions. Some partial results (Lhote 2015) for NFT.

Thank you.