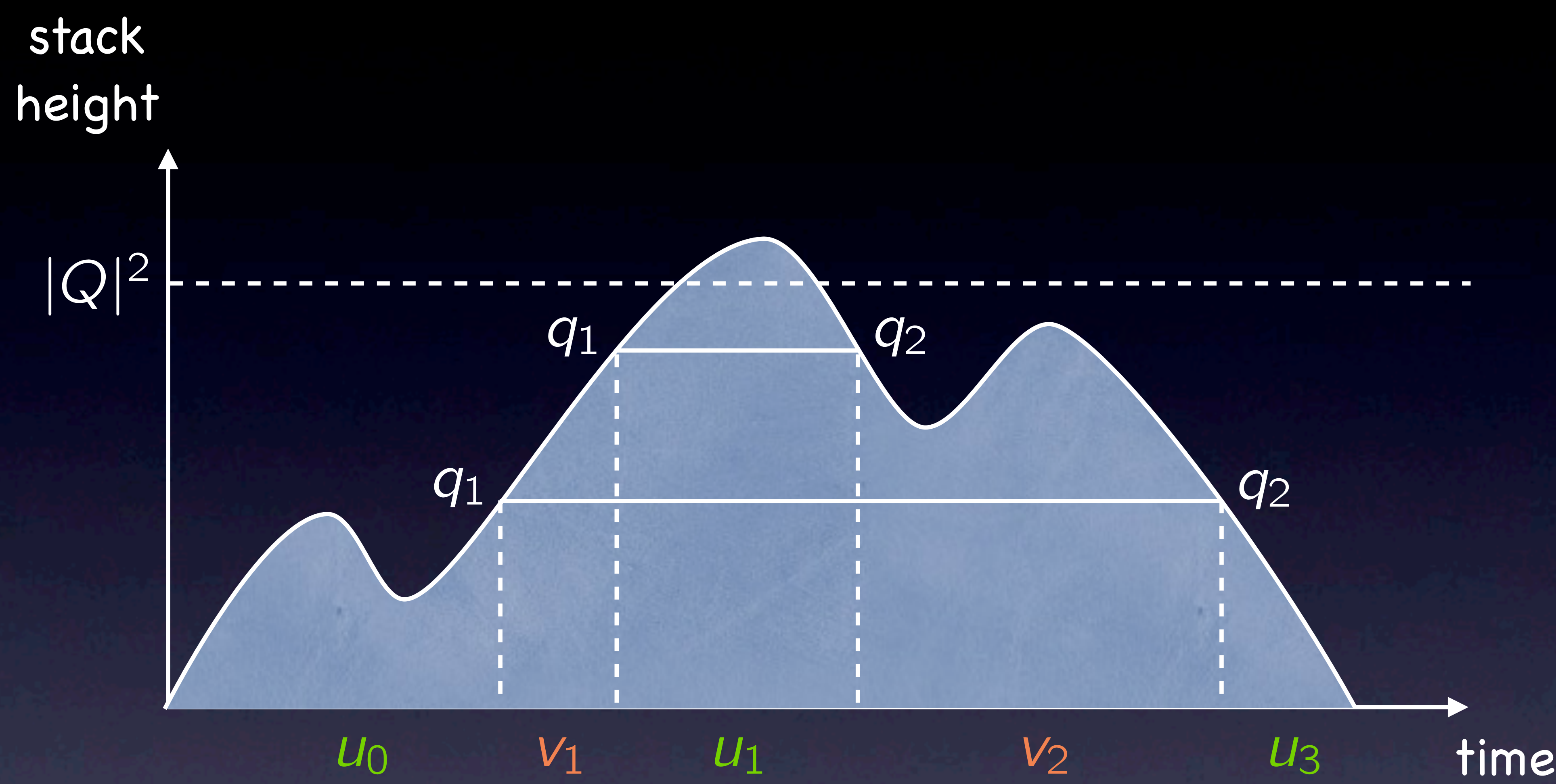
Pumping Lemma for m-MCFL$_{\mathrm{wn}}$

Theorem (Kanazawa 2009)**.**
$L \in m$-MCFL$_{\mathrm{wn}} \Rightarrow L$ is $2m$-iterative.

The proof of the Pumping Lemma for m–MCFL$_{\mathrm{wn}}$ is more complex.

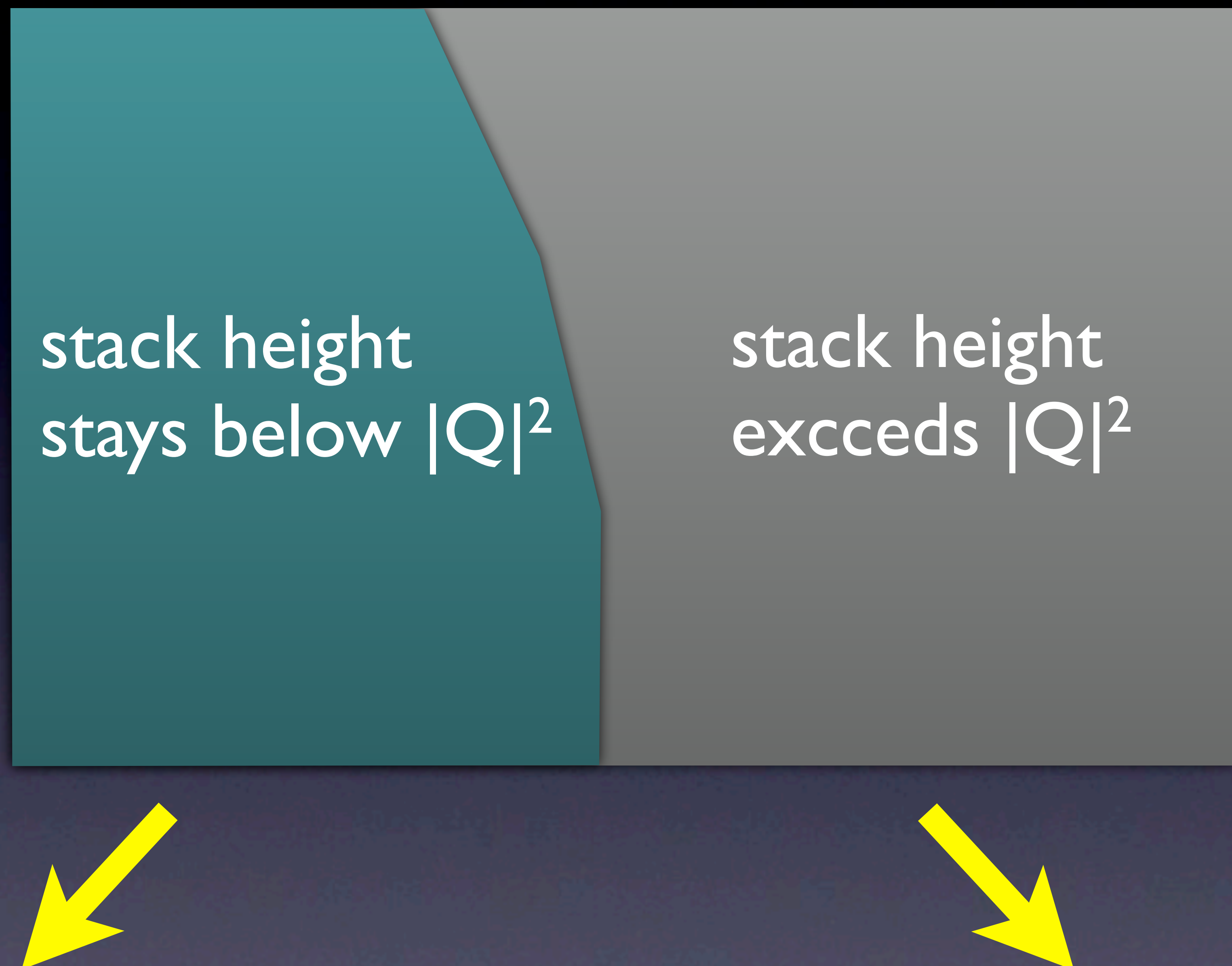
Pumping Lemma for PDA



- ¬(All but finitely many accepting computations reach stack height $|Q|^2$)
- { w | w has an accepting computation that doesn't reach stack height $|Q|^2$ } is regular

The proof in each case is somewhat similar to the proof of the pumping lemma for CFLs using PDA, rather than CFG.

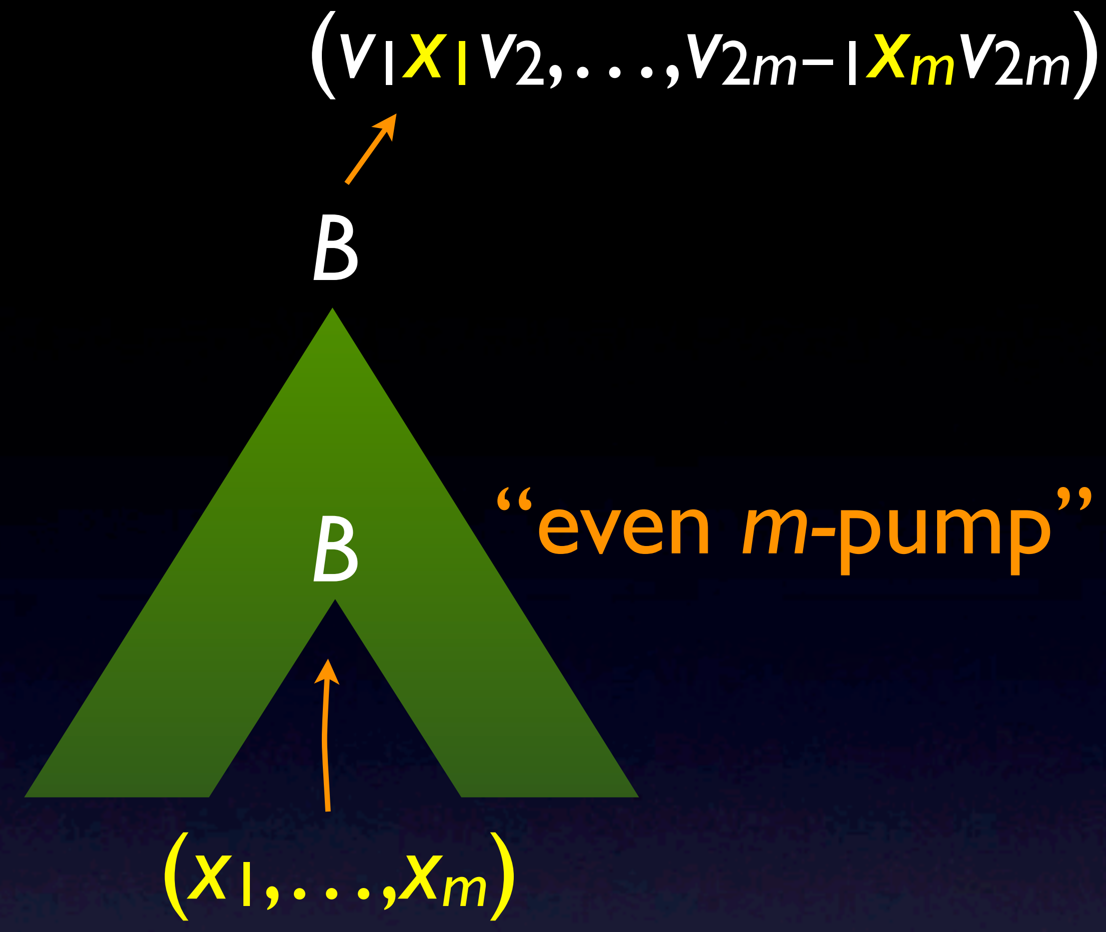
set of accepting computations of PDA M



accepts a regular set R

may be infinite

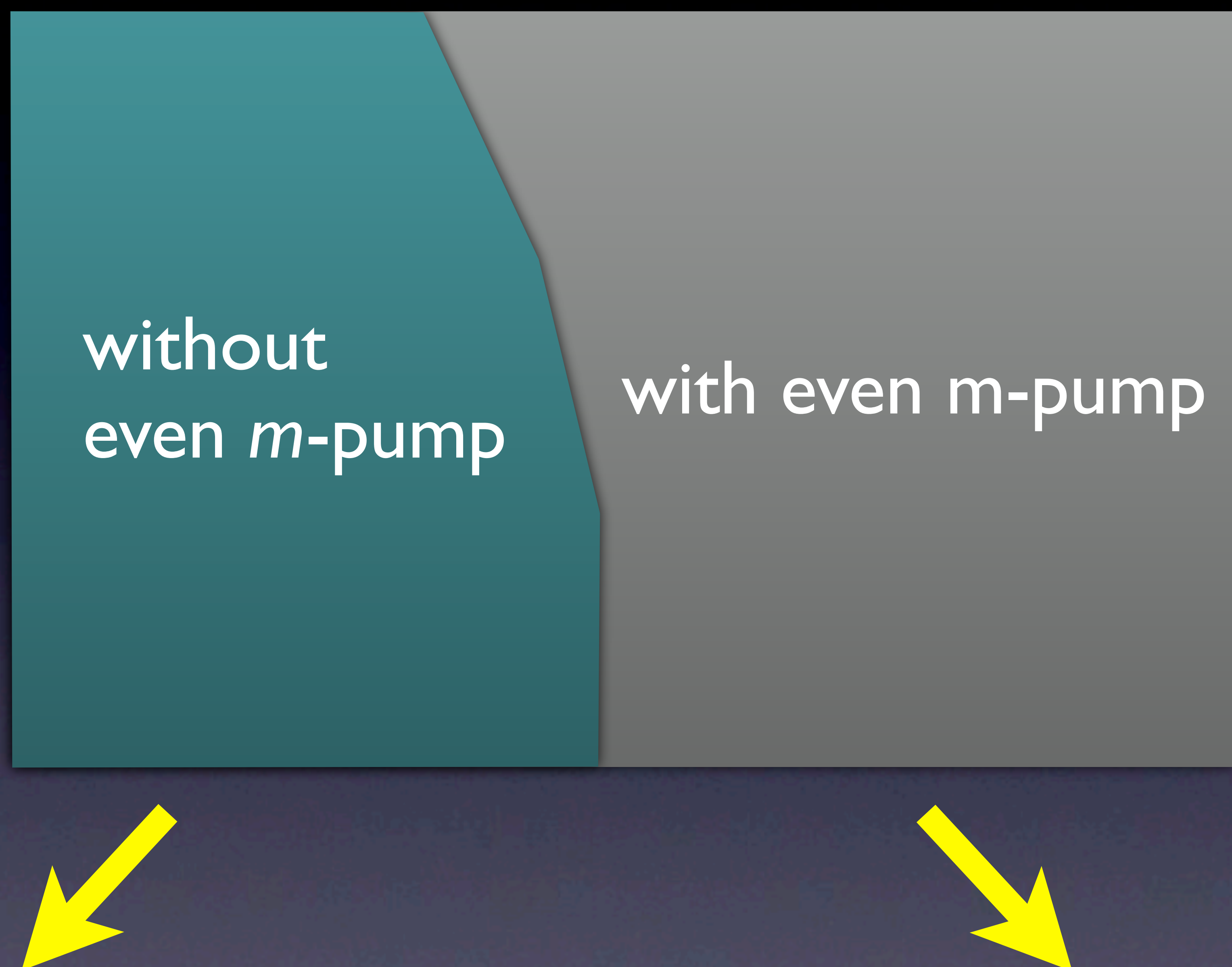
accepts a 2-iterative set



w has a derivation tree with even m-pump
$\Rightarrow$ w is $2m$-pumpable

$$w = u_0\ v_1\ u_1\ \ldots\ v_{2m}\ u_{2m}$$
$$v_1\ \ldots\ v_{2m} \neq \varepsilon$$
$$u_0\ v_1^{\,n}\ u_1\ \ldots\ v_{2m}^{\,n}\ u_{2m} \in L(G) \text{ for all } n \geq 0$$

set of derivation trees of $G \in m$-MCFG$_{wn}$

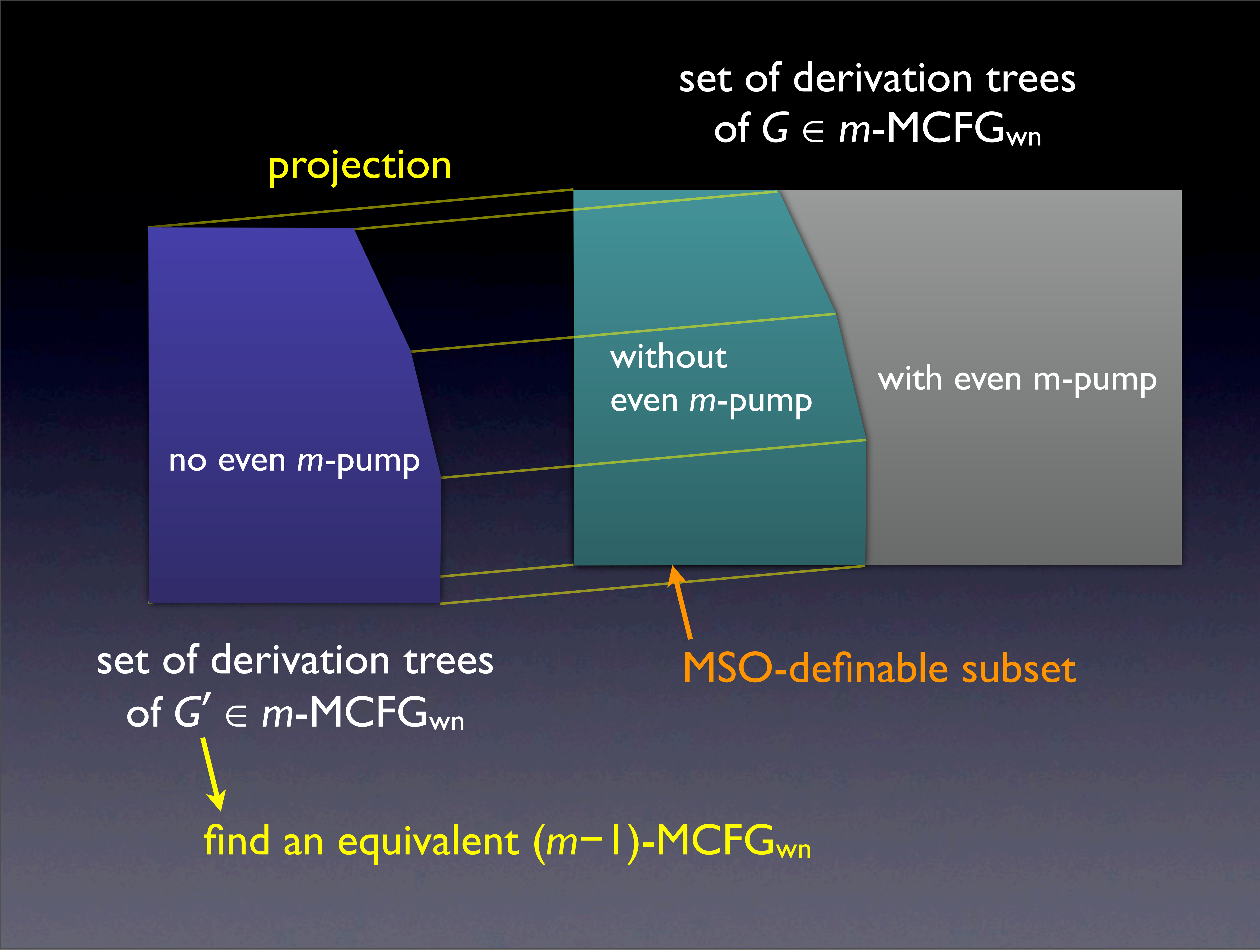


yields $L' \in (m-1)$-MCFL$_{wn}$

may be infinite

yields a $2m$-iterative set

If the derivation tree contains an even m-pump, the string is 2m-pumpable.
Otherwise, the string is in the language of some w.n. (m-1)-MCFG, and therefore is 2(m-1)-pumpable (disregarding finitely many exceptions).
Proof by induction on m.


set of derivation trees
of $G \in m$-MCFG$_{wn}$
projection
without
even m-pump
with even m-pump
no even m-pump
set of derivation trees
of $G' \in m$-MCFG$_{wn}$
MSO-definable subset
find an equivalent $(m-1)$-MCFG$_{wn}$

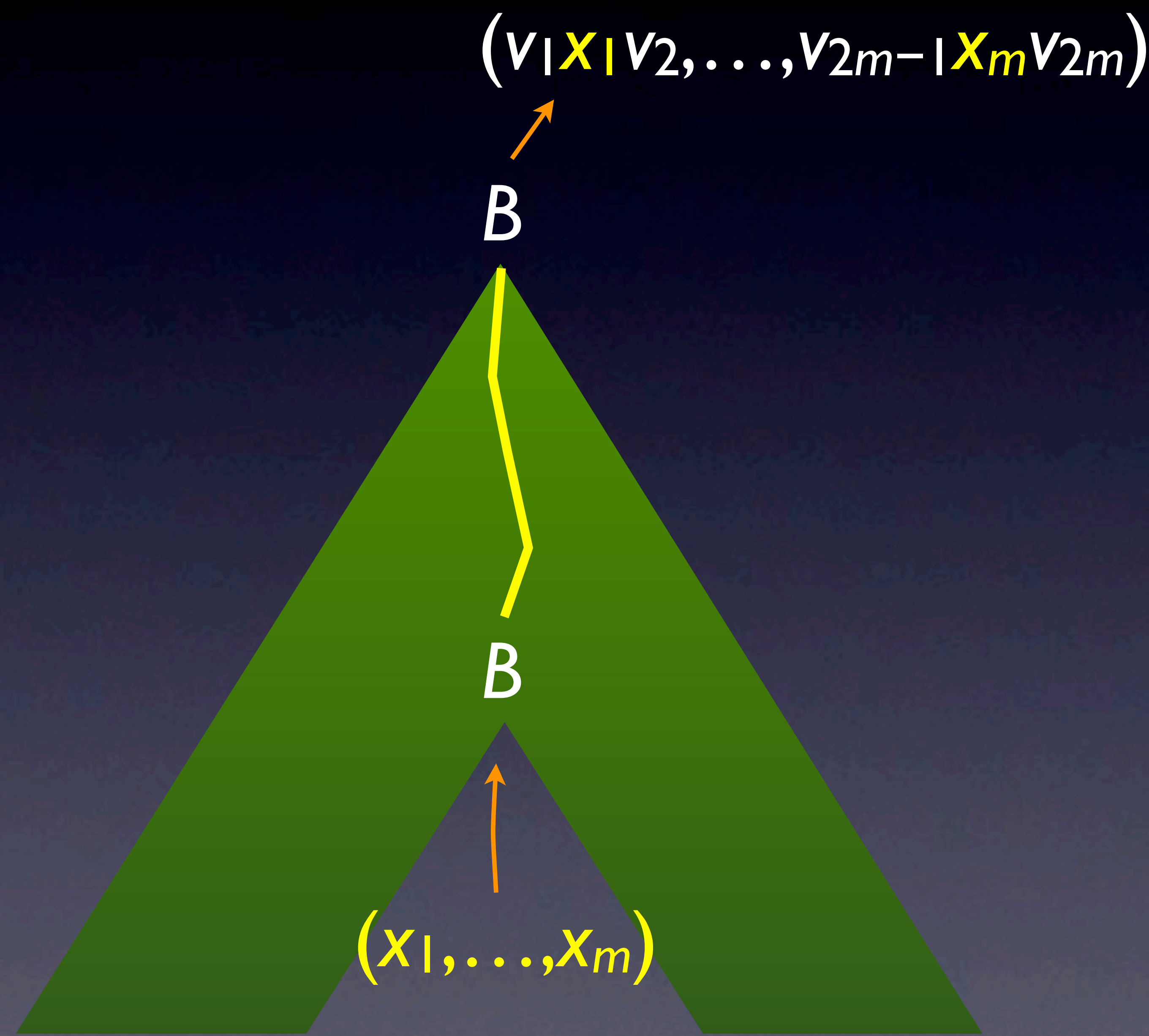
m-proper rules

$$B(\mathbf{x}_1\mathbf{y}_1\mathbf{z}_1, \mathbf{z}_2\mathbf{y}_2a, \mathbf{y}_3bc\mathbf{x}_2d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2), B(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3), C(\mathbf{z}_1, \mathbf{z}_2)$$

$m = 3$

m-proper on the second subgoal

The "spine" of an even *m*-pump consists of *m*-proper rules



m-degree

- *m*-degree of a rule is 0 if dimention of left-hand side nonterminal < *m*
- otherwise *m*-degree = number of right-hand side nonterminals of dimension *m*

$$B(\mathbf{x}_1\mathbf{y}_1\mathbf{x}_2,\ \mathbf{z}_1\mathbf{y}_2 a\mathbf{y}_3 b,\ c\mathbf{z}_2 d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2), B(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3), C(\mathbf{z}_1, \mathbf{z}_2) \qquad m = 3$$

m-degree = 1

Program Transformation

m-MCFG$_{wn}$ with no even m-pumps

↓

no m-proper rules

↓

total m-degree = 0

↓

$(m-1)$-MCFG$_{wn}$

The proof of this claim is by successive transformations on the grammar.

$m = 2$

π_1: $S(\mathbf{x}_1\mathbf{x}_2) \leftarrow B(\mathbf{x}_1, \mathbf{x}_2)$

π_2: $B(a\mathbf{x}_1 b, c\mathbf{x}_2 d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2)$

m-proper rule

π_3: $A(a\mathbf{x}_1 b\mathbf{x}_2 c, d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2)$

π_4: $A(\varepsilon, \varepsilon) \leftarrow$

$\downarrow$ unfolding

π_1: $S(\mathbf{x}_1\mathbf{x}_2) \leftarrow B(\mathbf{x}_1, \mathbf{x}_2)$

$\pi_2 \circ \pi_3$: $B(aa\mathbf{x}_1 b\mathbf{x}_2 cb, cdd) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2)$

$\pi_2 \circ \pi_4$: $B(ab, cd) \leftarrow$

π_3: $A(a\mathbf{x}_1 b\mathbf{x}_2 c, d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2)$

π_4: $A(\varepsilon, \varepsilon) \leftarrow$

A rule is m-proper if the head nonterminal is m-ary and there is an m-ary nonterminal on the right-hand side, each of whose arguments appear in the corresponding argument of the head nonterminal.
Unfold until there is no m-proper rule. This procedure terminates because the grammar does not allow an even m-pump.

π_1: $S(\boldsymbol{x}_1\boldsymbol{x}_2) \leftarrow B(\boldsymbol{x}_1, \boldsymbol{x}_2)$

π_5: $B(aa\boldsymbol{x}_1b\boldsymbol{x}_2cb, cdd) \leftarrow A(\boldsymbol{x}_1, \boldsymbol{x}_2)$ m-degree = 1

π_6: $B(ab, cd) \leftarrow$

π_3: $A(a\boldsymbol{x}_1b\boldsymbol{x}_2c, d) \leftarrow A(\boldsymbol{x}_1, \boldsymbol{x}_2)$ m-degree = 1

π_4: $A(\varepsilon, \varepsilon) \leftarrow$

$\downarrow$ unfolding^{-1}

π_1: $S(\boldsymbol{x}_1\boldsymbol{x}_2) \leftarrow B(\boldsymbol{x}_1, \boldsymbol{x}_2)$

$\pi_{5.1}$: $B(aa\boldsymbol{x}cb, cdd) \leftarrow C(\boldsymbol{x})$

$\pi_{5.2}$: $C(\boldsymbol{x}_1b\boldsymbol{x}_2) \leftarrow A(\boldsymbol{x}_1, \boldsymbol{x}_2)$

$\pi_5 = \pi_{5.1} \circ \pi_{5.2}$

π_2: $B(ab, cd) \leftarrow$

$\pi_{3.1}$: $A(a\boldsymbol{x}c, d) \leftarrow D(\boldsymbol{x})$

$\pi_{3.2}$: $D(\boldsymbol{x}_1b\boldsymbol{x}_2) \leftarrow A(\boldsymbol{x}_1, \boldsymbol{x}_2)$

$\pi_3 = \pi_{3.1} \circ \pi_{3.2}$

π_4: $A(\varepsilon, \varepsilon) \leftarrow$

The m-degree of a rule is 0 if the arity of the head nonterminal is < m; otherwise it's the number of m-ary nonterminals on the right-hand side.
Do the converse of unfolding.

π_1: $S(\boldsymbol{x}_1\boldsymbol{x}_2) \leftarrow B(\boldsymbol{x}_1, \boldsymbol{x}_2)$

$\pi_{5.1}$: $B(aa\boldsymbol{x}cb, cdd) \leftarrow C(\boldsymbol{x})$

$\pi_{5.2}$: $C(\boldsymbol{x}_1 b\boldsymbol{x}_2) \leftarrow A(\boldsymbol{x}_1, \boldsymbol{x}_2)$

π_2: $B(ab, cd) \leftarrow$

$\pi_{3.1}$: $A(a\boldsymbol{x}c, d) \leftarrow D(\boldsymbol{x})$

$\pi_{3.2}$: $D(\boldsymbol{x}_1 b\boldsymbol{x}_2) \leftarrow A(\boldsymbol{x}_1, \boldsymbol{x}_2)$

π_4: $A(\varepsilon, \varepsilon) \leftarrow$

$\downarrow$ unfolding

$\pi_1 \circ \pi_{5.1}$: $S(aa\boldsymbol{x}cbcdd) \leftarrow C(\boldsymbol{x})$

$\pi_1 \circ \pi_2$: $S(abcd) \leftarrow$

$\pi_{5.2} \circ \pi_{3.1}$: $C(a\boldsymbol{x}cbd) \leftarrow D(\boldsymbol{x})$

$\pi_{5.2} \circ \pi_4$: $C(b) \leftarrow$

$\pi_{3.2} \circ \pi_{3.1}$: $D(a\boldsymbol{x}cbd) \leftarrow D(\boldsymbol{x})$

$\pi_{3.2} \circ \pi_4$: $D(b) \leftarrow$

Now each rule contains m-ary nonterminals only on one side of the rule, if any. Unfolding eliminates all m-ary nonterminals.

Program Transformation

m-MCFG$_{wn}$ with no even m-pumps

↓ unfolding

no m-proper rules

↓ unfolding^{-1}

total m-degree = 0

↓ unfolding

$(m-1)$-MCFG$_{wn}$

Reduction of *m*-degrees

$m = 3$

$$B(\mathbf{x}_1\mathbf{y}_1\mathbf{z}_1, \mathbf{z}_2\mathbf{y}_2 a \mathbf{y}_3 b, c\mathbf{x}_2 d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2), B(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3), C(\mathbf{z}_1, \mathbf{z}_2)$$

$\downarrow$ unfolding^{-1}

$$B(\mathbf{x}_1\mathbf{w}_1, \mathbf{w}_2 b, c\mathbf{x}_2 d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2), D(\mathbf{w}_1, \mathbf{w}_2)$$

$$D(\mathbf{y}_1\mathbf{z}_1, \mathbf{z}_2\mathbf{y}_2 a \mathbf{y}_3) \leftarrow B(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3), C(\mathbf{z}_1, \mathbf{z}_2)$$

The well-nestedness assumption is necessary in the second step.
Here’s a case of a well-nested rule.

Reduction of *m*-degrees

$m = 3$

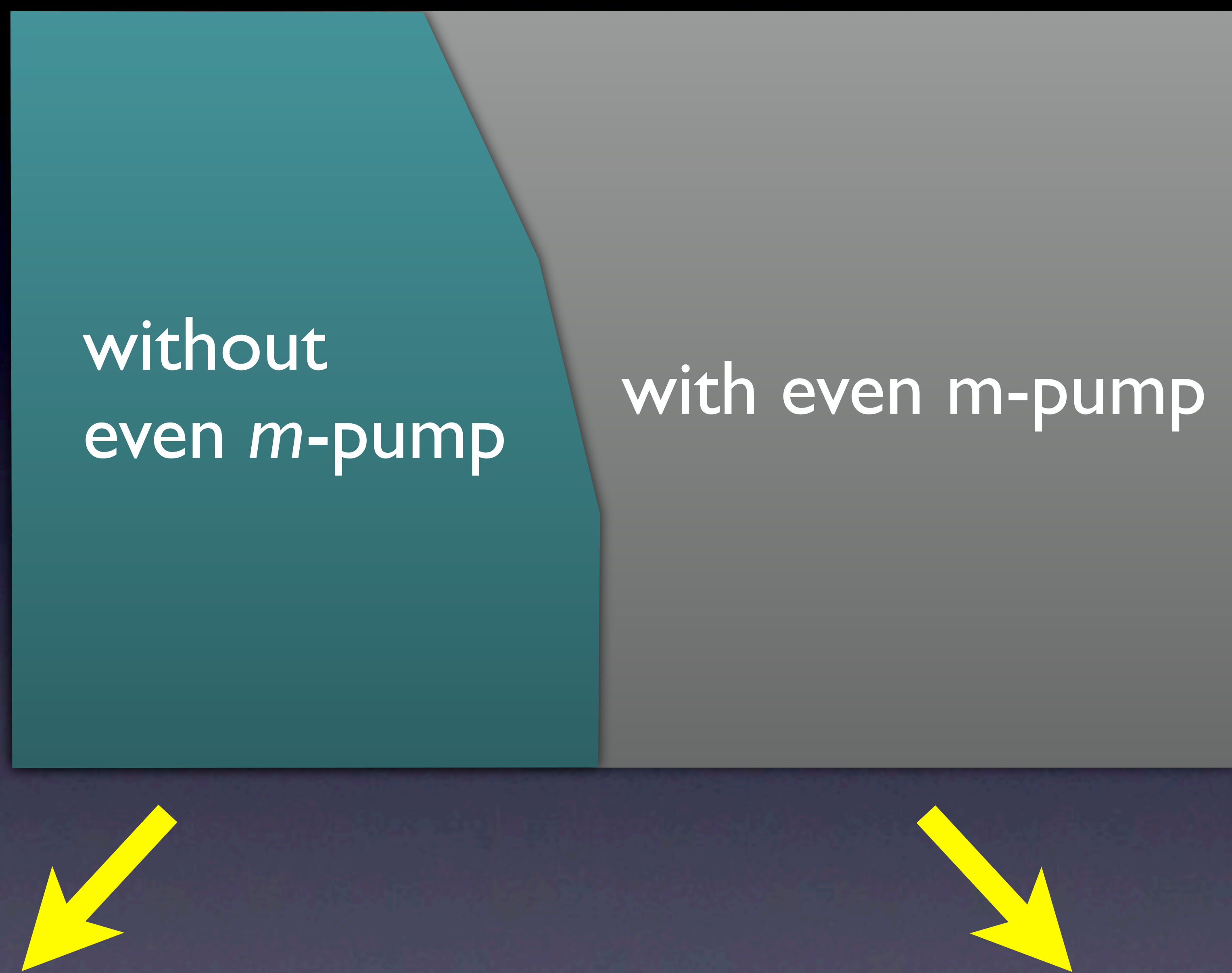
$$B(\mathbf{x}_1\mathbf{y}_1\mathbf{x}_2,\ \mathbf{z}_1\mathbf{y}_2a\mathbf{y}_3b,\ c\mathbf{z}_2d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2), B(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3), C(\mathbf{z}_1, \mathbf{z}_2)$$

$\downarrow$ unfolding^{-1}

×

If a rule is non-well-nested, the procedure does not work.

set of derivation trees of $G \in m\text{-MCFG}_{wn}$



yields $L' \in (m-1)\text{-MCFL}_{wn}$

may be infinite

yields a $2m$-iterative set

If the derivation tree contains an even m-pump, the string is 2m-pumpable.
Otherwise, the string is in the language of some w.n. (m-1)-MCFG, and therefore is 2(m-1)-pumpable (disregarding finitely many exceptions).
Proof by induction on m.

Pumping Lemma for m-MCFL$_{wn}$

Theorem (Kanazawa 2009)**.**
$L \in m\text{-MCFL}_{wn} \Rightarrow L$ is $2m$-iterative.

The proof of the Pumping Lemma for m-MCFL$_{wn}$ is more complex.

Reduction of *m*-degrees

$m = 2$

$B(\mathbf{x}_1\mathbf{y}_1\mathbf{x}_2\mathbf{y}_2 b, \mathbf{z}_1 c) \leftarrow$
$A(\mathbf{x}_1, \mathbf{x}_2), B(\mathbf{y}_1, \mathbf{y}_2), C(\mathbf{z}_1)$

$\downarrow$ unfolding^{-1}

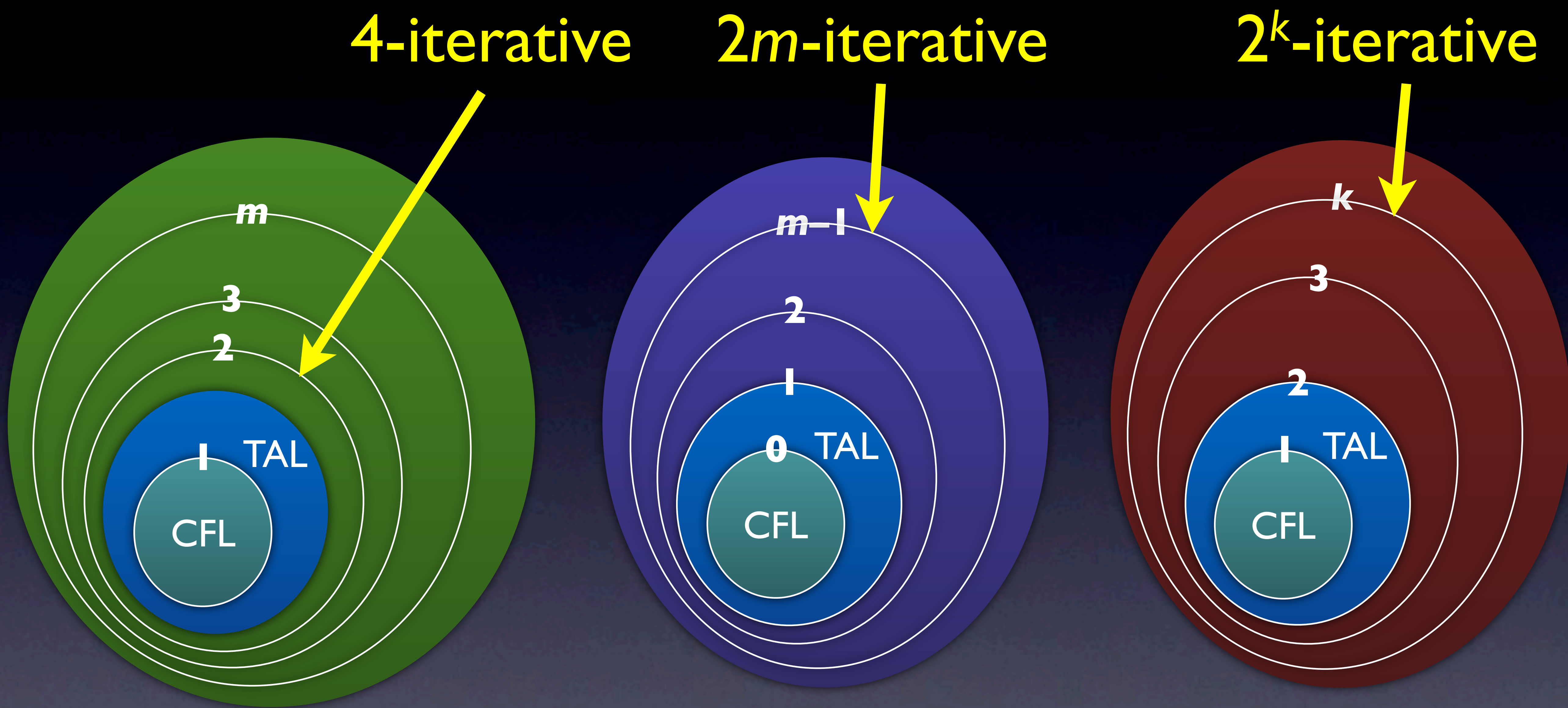
$B(\mathbf{w}_1 b, \mathbf{z}_1 c) \leftarrow D(\mathbf{w}_1), C(\mathbf{z}_1)$
$D(\mathbf{x}_1\mathbf{y}_1\mathbf{x}_2\mathbf{y}_2) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2), B(\mathbf{y}_1, \mathbf{y}_2)$

For m = 2, the procedure works even when the rule is non-well-nested.

Pumping Lemma for 2-MCFL

Theorem (Kanazawa 2009)**.**
$L \in$ 2-MCFL $\Rightarrow L$ is 4-iterative.

The Pumping Lemma holds for 2-MCFL.



$$\mathrm{MCFL} = \bigcup_{m \geq 1} m\text{-}\mathrm{MCFL} \qquad y\mathrm{CFT}_{\mathrm{sp}} = \bigcup_{m \geq 1} y\mathrm{CFT}_{\mathrm{sp}}(m-1) \qquad \mathbf{C} = \bigcup_{k \geq 1} \mathbf{C}_k$$

$$\mathrm{MCFL}_{\mathrm{wn}} = \bigcup_{m \geq 1} m\text{-}\mathrm{MCFL}_{\mathrm{wn}}$$

The proof shows that a 2-MCFL is 4-iterative.

Counterexample in 3-MCFL

$H(\mathbf{x}_2) \leftarrow G(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$

$G(a\mathbf{x}_1, \mathbf{y}_1 c\mathbf{x}_2\bar{c}d\mathbf{y}_2\bar{d}\mathbf{x}_3, \mathbf{y}_3 b) \leftarrow G(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3), G(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3)$

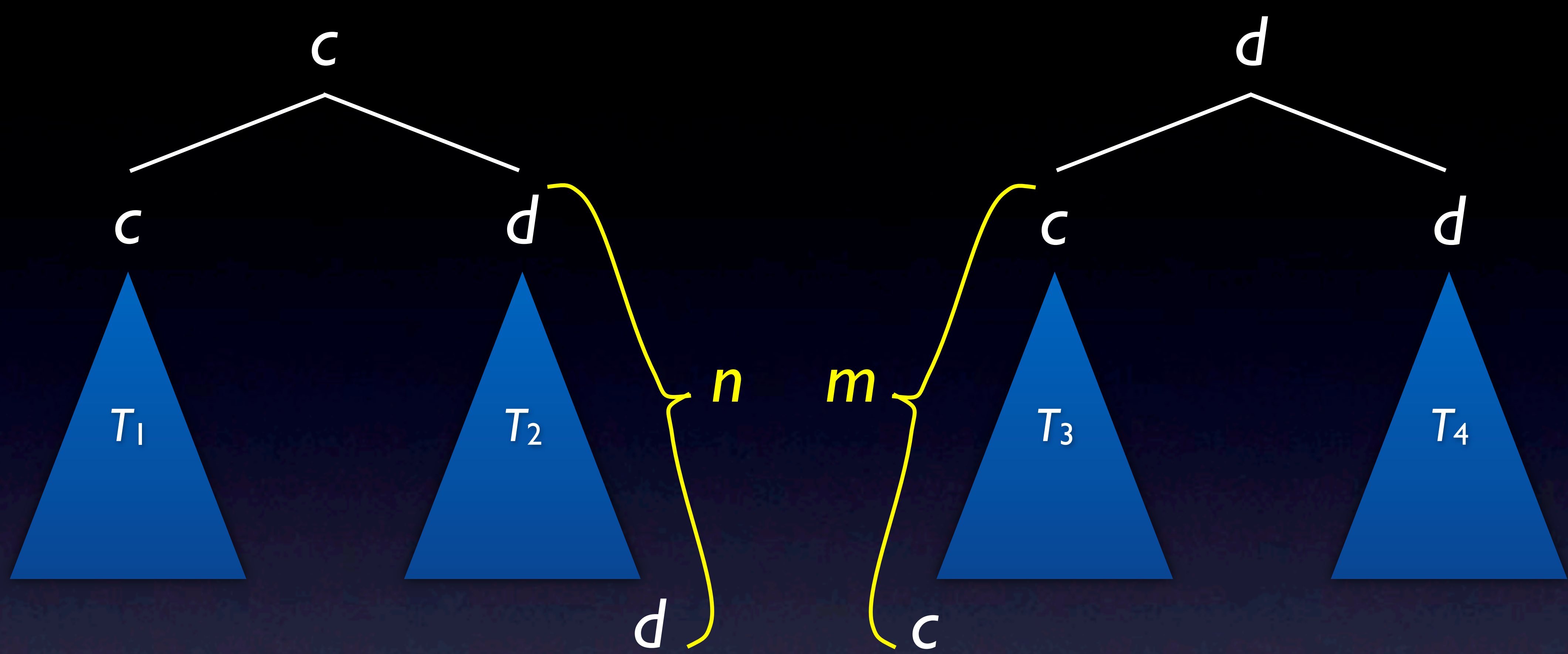
$G(a, \varepsilon, b) \leftarrow$

$$\varphi:\ H \xrightarrow[\text{bijection}]{a := \varepsilon,\, b := \varepsilon} V$$

$V \rightarrow \varepsilon \mid cV\bar{c}\,dV\bar{d}$ $\qquad$ $V \subseteq D_2^*$

Pumping fails for m-MCFLs for (m > 2).
Here's an example of a 3-MCFL that is not k-iterative for any k.

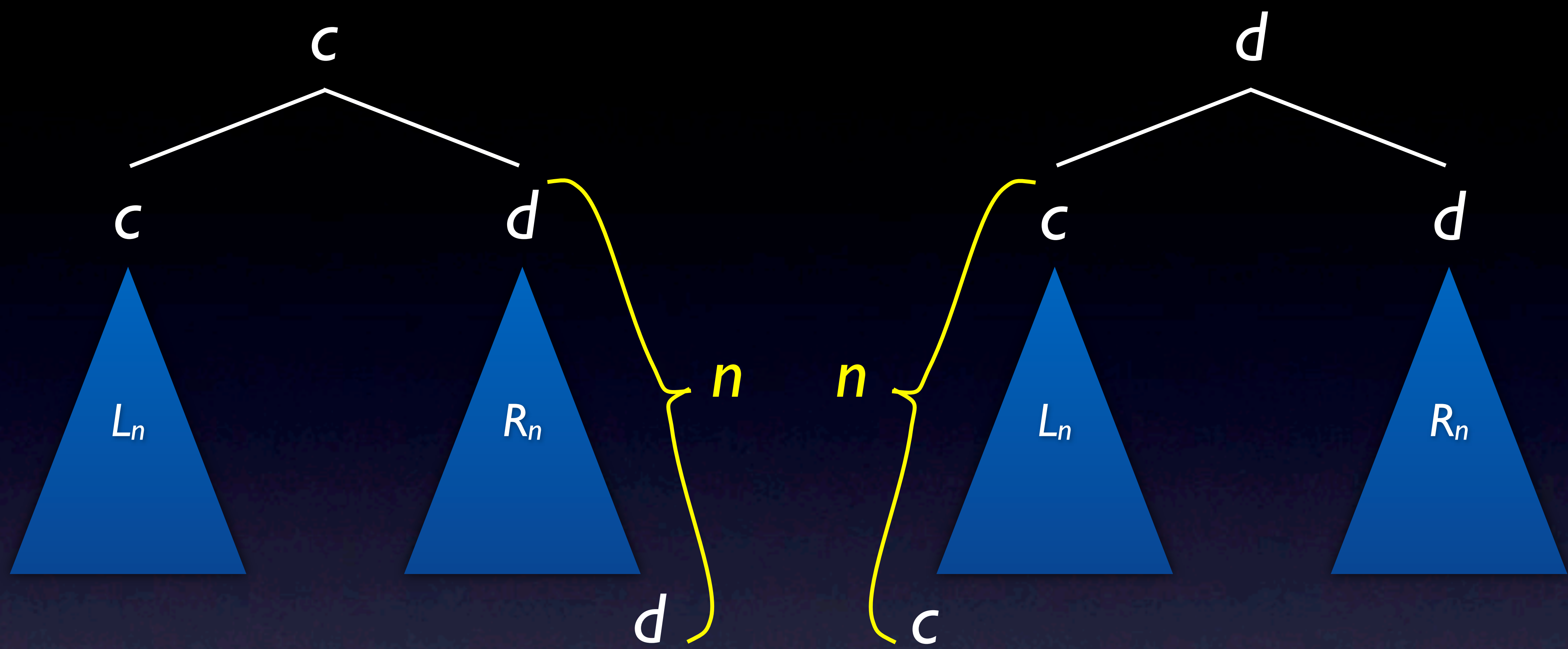
tree representation of D_2^*



$$\Big\downarrow \varphi^{-1}$$

$$a^{m+1}\; c\; \varphi^{-1}(T_1\ T_2)\; \overline{c}\; d\; \varphi^{-1}(T_3\ T_4)\; \overline{d}\; b^{n+1}$$

perfect binary tree



$L_1 = c, R_1 = d$

φ^{-1}

$$v_{n+1} = a^{n+1} c\, v_n\, \overline{c}\, d\, v_n\, \overline{d}\, b^{n+1} \qquad v_0 = \varepsilon$$

not pumpable

Lemma.

$$w \in \mathrm{fac}(\{\, v_n \mid n \geq 0 \,\}) \wedge ww \in \mathrm{fac}(H)$$

$$\downarrow$$

$$\varphi(w) \in c^* \mid cV\bar{c}dc^* \mid \bar{d}^* \mid \bar{d}^*\bar{c}dV\bar{d} \mid V\bar{c}dc^+ \mid \bar{d}^+\bar{c}dV$$

$v \in H$ is said to be almost anti-iterative if

$v = u_0\, w_1\, u_1 \ldots w_k\, u_k \wedge w_1 \ldots w_k \neq \varepsilon$ for any $k \geq 1$

implies

there is at most one $i > 1$ such that

$u_0\, w_1^{\,i}\, u_1 \ldots w_k^{\,i}\, u_k \in H$

Theorem. For each $n \geq 0$, v_n is almost anti-iterative.

$$v_2 = \underbrace{aac}_{w_1}\ \underbrace{ac\bar{c}d\bar{d}}_{u_1}\ \underbrace{b\bar{c}dac\bar{c}d\bar{d}b\bar{d}b}_{w_2}\ \underbrace{b}_{w_3}$$

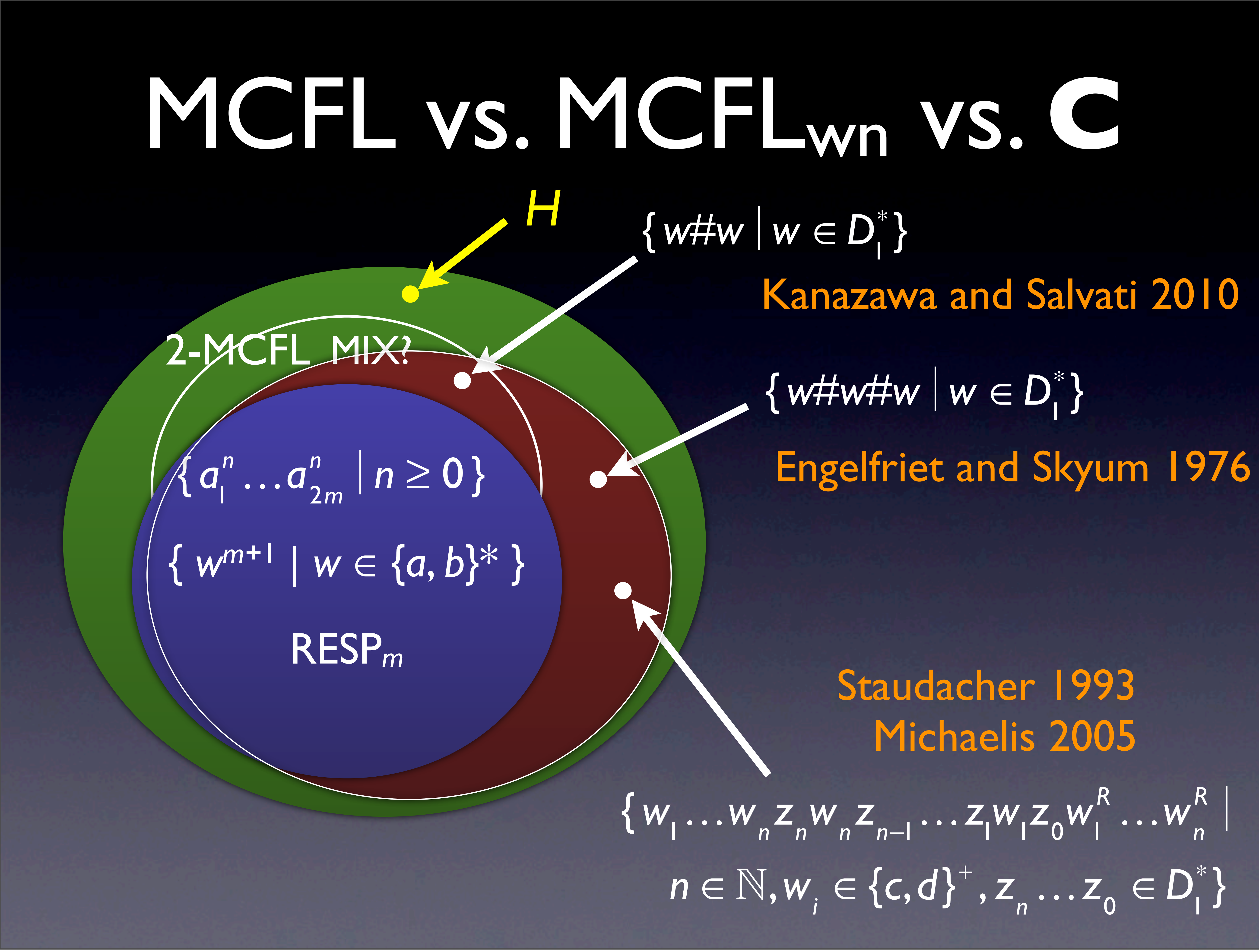
$$w_1^2u_1w_2^2w_3^2 = \ aac\ \underbrace{aac\ \underbrace{ac\bar{c}d\bar{d}\ b}_{v_1}\bar{c}d\,\underbrace{ac\bar{c}d\bar{d}b}_{v_1}\bar{d}b\ b}_{v_2}\,\bar{c}d\,\underbrace{ac\bar{c}d\bar{d}b}_{v_1}\bar{d}b\ b\ b\ \in H$$

$$w_1^3u_1w_2^3w_3^3 =$$

$$aac\ \underbrace{aac\ \underbrace{aac\ \underbrace{ac\bar{c}d\bar{d}\ b}_{v_1}\bar{c}d\,\underbrace{ac\bar{c}d\bar{d}b}_{v_1}\bar{d}b\ b}_{v_2}\,\bar{c}d\,\underbrace{ac\bar{c}d\bar{d}b}_{v_1}\bar{d}b\ b}_{\notin H}\,\bar{c}d\,\underbrace{ac\bar{c}d\bar{d}b}_{v_1}\bar{d}b\ b\ b\ b\ \notin H$$

Corollary. There is a 3-MCFL that is not *k*-iterative for any *k*.

Kanazawa, Kobele, Michaelis, Salvati, & Yoshinaka 201x



Since every language in C is k-iterative for some k, this language separates MCFL from C.

Limited Cross-Serial Dependencies

"MCSGs capture only certain kinds of dependencies, such as nested dependencies and certain limited kinds of crossing dependencies (for example, in subordinate clause constructions in Dutch or some variations of them, but perhaps not in the so-called MIX … language …)"

Joshi, Vijay-Shanker, and Weir 1991

$$\mathrm{MIX} = \{\, w \in \{a,b,c\}^* \mid |w|_a = |w|_b = |w|_c \,\}$$

The language MIX was supposed to be outside of the class of mildly context-sensitive languages.

“[No human language] has … complete freedom for order.” Bach 1981

“[MIX represents] an extremely case of the degree of free word order permitted in a language … which is linguistically not relevant.” Joshi 1985

“… it seems rather unlikely that any natural language will turn out to have a MIX-like characteristic.” Gazdar 1985

“TAGs cannot generate this language, although for TAGs the proof is not in hand yet.” Joshi 1985

“I conjecture that this language is not indexed.”
Marsh 1986

“It is not known whether TAG … can generate MIX. This has turned out to be a very difficult problem.”
Joshi et al. 1991

Some Facts about MIX

rational transduction

$$L_1 \geq L_2 \Leftrightarrow \exists g\ h\ R\ (L_2 = g(h^{-1}(L_1) \cap R))$$

g, h: homomorphisms
R: regular set

$$L_1 \equiv L_2 \Leftrightarrow L_1 \geq L_2 \wedge L_1 \leq L_2$$

rationally equivalent

$\mathcal{L}$ is a rational cone $\Leftrightarrow$ $\mathcal{L}$ is closed under $\geq$

Some Facts about MIX

$$\text{MIX} \equiv O_2$$

$$O_2 = \{\, w \in \{a, \overline{a}, b, \overline{b}\}^* \mid |w|_a = |w|_{\overline{a}}, |w|_b = |w|_{\overline{b}} \,\}$$

$$g(O_2 \cap (a \cup b \cup \overline{ab})^*) = \text{MIX}$$

$$g(a) = a$$

$$g(b) = b$$

$$g(\overline{a}) = c$$

$$g(\overline{b}) = \varepsilon$$

$$O_2 \geq \text{MIX}$$

$$h^{-1}(\mathrm{MIX} \cap (a^2 \cup (ab)^2 \cup (bc)^2 \cup c^2)^*) = O_2$$

$$h(a) = a^2$$

$$h(\overline{a}) = (bc)^2$$

$$h(b) = (ab)^2$$

$$h(\overline{b}) = c^2$$

$\mathrm{MIX} \geq O_2$

Combinatorial Group Theory

"… it does not … seem to be known whether or not the word problem of $Z \times Z$ is indexed." Gilman 2005

O_2

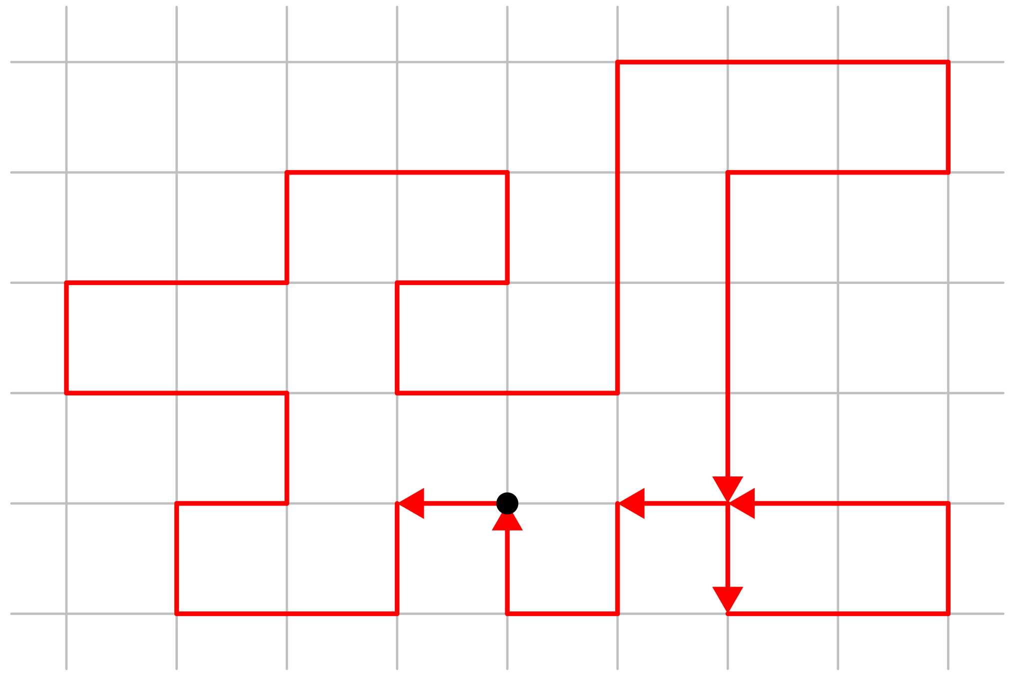
Theorem (Salvati). $O_2 \in$ 2-MCFL

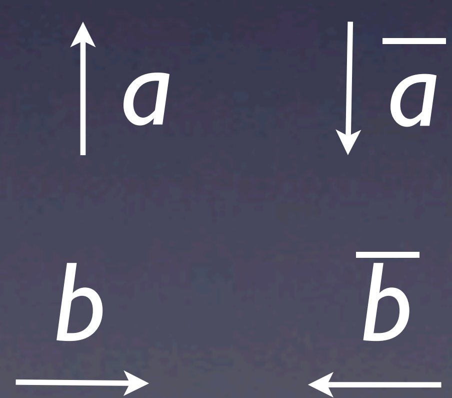
$$
\begin{array}{rcl}
S(xy) & \leftarrow & Inv(x, y) \\
\hline
Inv(x_1 y_1, y_2 x_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
Inv(x_1 x_2 y_1, y_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
Inv(y_1, x_1 x_2 y_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
Inv(y_1 x_1 x_2, y_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
Inv(y_1, y_2 x_1 x_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
\hline
Inv(\alpha x_1 \overline{\alpha}, x_2) & \leftarrow & Inv(x_1, x_2) \\
Inv(\alpha x_1, \overline{\alpha} x_2) & \leftarrow & Inv(x_1, x_2) \\
Inv(\alpha x_1, x_2 \overline{\alpha}) & \leftarrow & Inv(x_1, x_2) \\
Inv(x_1 \alpha, \overline{\alpha} x_2) & \leftarrow & Inv(x_1, x_2) \\
Inv(x_1 \alpha, x_2 \overline{\alpha}) & \leftarrow & Inv(x_1, x_2) \\
Inv(x_1, \alpha x_2 \overline{\alpha}) & \leftarrow & Inv(x_1, x_2) \\
\hline
Inv(x_1 y_1 x_2, y_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
Inv(x_1, y_1 x_2 y_2) & \leftarrow & Inv(x_1, x_2), Inv(y_1, y_2) \\
\hline
Inv(\epsilon, \epsilon) & \leftarrow &
\end{array}
$$

where $\alpha \in \{a; b\}$

Theorem: Given w_1 and w_2 such that $w_1 w_2 \in O_2$, $Inv(w_1, w_2)$ is derivable.

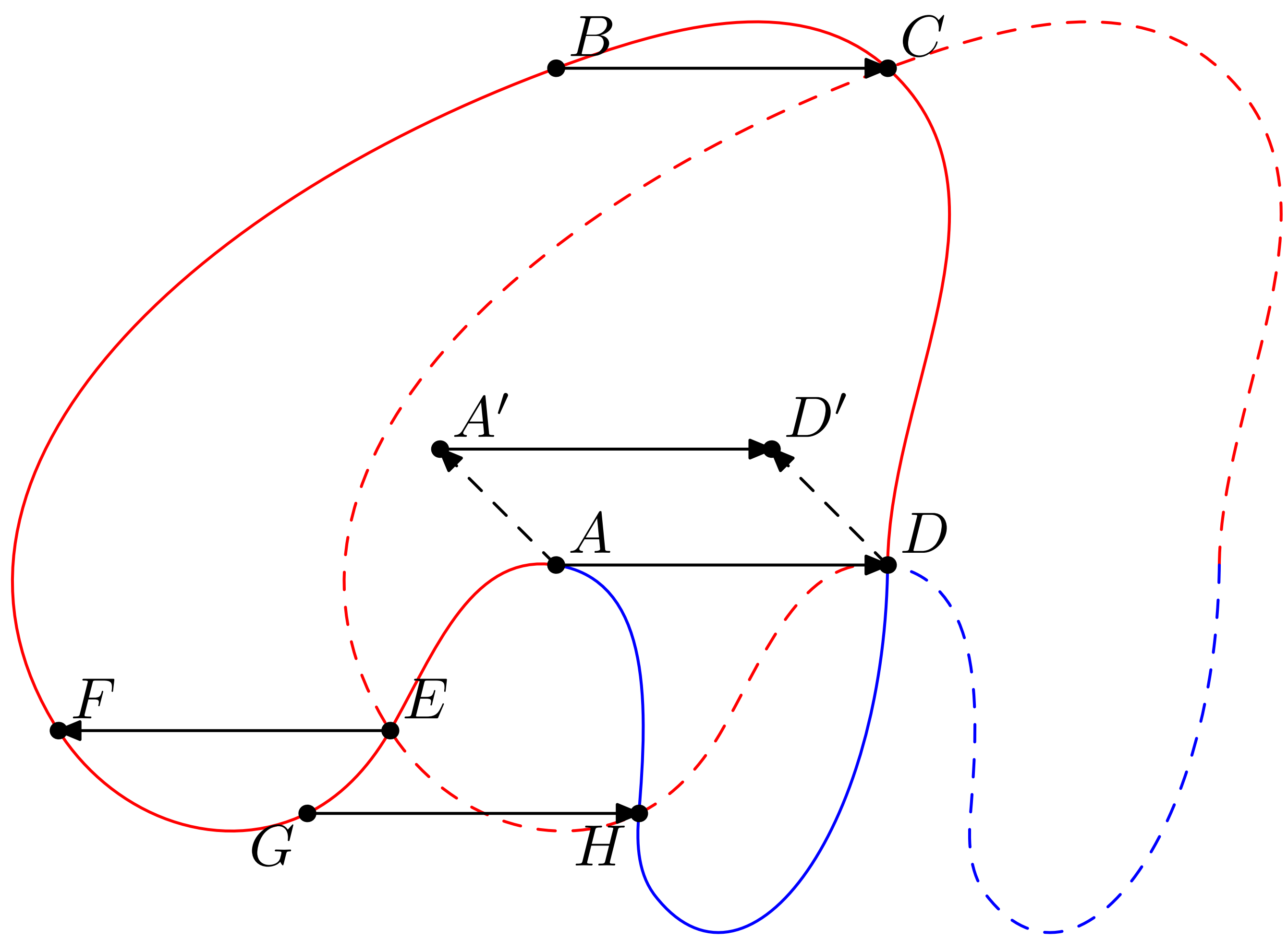
The words in O_2 are precisely the words that are represented as closed curves:
$\bar{b}\bar{a}\bar{b}\bar{b}aba\bar{b}\bar{b}abbabb\bar{a}\bar{b}\bar{a}bbaaabbb\bar{a}\bar{b}\bar{b}\bar{a}\bar{a}\bar{a}\bar{a}bba\bar{b}\bar{b}\bar{b}\bar{a}ba$





A theorem on Jordan curves

Theorem: If A and D are two points on a Jordan curve J such that there are two points A' and D' inside J such that $\overrightarrow{AD} = \overrightarrow{A'D'}$, then there are two points B and C pairwise distinct from A and D such that A, B, C, and D appear in that order on one of the arcs going from A to D and $\overrightarrow{AD} = \overrightarrow{BC}$.



two-sided Dyck language

$$\mathrm{MIX} \equiv O_2 = \mathrm{Shuffle}(\hat{D}_1^*, \hat{D}_1^*)$$

Fact. $\mathrm{Shuffle}(D_1^*, D_1^*) \notin$ 2-MCFL

The one-sided analogue of O2 (the set of curves within the first quadrant) is not a 2-MCFL. This can be proved with the Pumping Lemma for 2-MCFL (Kanazawa 2009).

$$\mathrm{MIX}_k = \{ w \in \{ a_1,\ldots,a_k \}^* \mid \psi(w) = n \cdot (1,\ldots,1) \}$$

$\mathrm{MIX} = \mathrm{MIX}_3$

Fact. If $\mathcal{L}$ is a rational cone and contains MIX_k for all k, then $\mathcal{L}$ contains all commutative semilinear languages.

Generalization of MIX.

Fact.

$\mathrm{MIX} \notin 2\text{-MCFL}(1)$

$\mathrm{MIX}_4 \notin 2\text{-MCFL}_{wn}$

$\mathrm{MIX}_{k+1} \notin k\text{-MCFL}(1)$

$\mathrm{MIX}_{k+2} \notin k\text{-MCFL}_{wn}$

Appropriate refinements of the Pumping Lemma for MCFLwn give these facts.

Question. $\mathrm{MIX}_4 \notin \mathrm{MCFL}$?

Question. $\mathrm{MIX} \notin \mathrm{MCFL}_{wn}$?

Currently have no idea how to prove these.

Theorem. MIX $\notin$ 2-MCFL$_{wn}$.

Kanazawa & Salvati 2012

Head Grammars

$$A(\boldsymbol{x}_1\boldsymbol{x}_2\boldsymbol{y}_1, \boldsymbol{y}_2) \leftarrow B(\boldsymbol{x}_1, \boldsymbol{x}_2), C(\boldsymbol{y}_1, \boldsymbol{y}_2)$$

left concatenation

$$A(\boldsymbol{x}_1, \boldsymbol{x}_2\boldsymbol{y}_1\boldsymbol{y}_2) \leftarrow B(\boldsymbol{x}_1, \boldsymbol{x}_2), C(\boldsymbol{y}_1, \boldsymbol{y}_2)$$

right concatenation

$$A(\boldsymbol{x}_1\boldsymbol{y}_1, \boldsymbol{y}_2\boldsymbol{x}_2) \leftarrow B(\boldsymbol{x}_1, \boldsymbol{x}_2), C(\boldsymbol{y}_1, \boldsymbol{y}_2)$$

wrapping

$$A(w_1, w_2) \leftarrow$$

$$w_1, w_2 \in \Sigma \cup \{\varepsilon\}$$

normal form for 2-MCFG$_{\mathrm{wn}}$

$$\begin{aligned} \psi_1(w) &= |w|_a - |w|_c, \\ \psi_2(w) &= |w|_b - |w|_c, \\ \psi(w) &= (\psi_1(w), \psi_2(w)). \end{aligned}$$

$$w \in \mathrm{MIX} \quad \text{iff} \quad \psi(w) = (0,0).$$

Lemma 2. *Suppose that $G = (N, \Sigma, P, S)$ is a head grammar without useless nonterminals such that $L(G) \subseteq MIX$. There exists a function $\Psi_G : N \to \mathbb{Z} \times \mathbb{Z}$ such that $\vdash_G A(u_1, u_2)$ implies $\psi(u_1 u_2) = \Psi_G(A)$.*

A *decomposition* of $w \in \Sigma^*$ is a finite binary tree satisfying the following conditions:

- the root is labeled by some (w_1, w_2) such that $w = w_1 w_2$,
- each internal node whose left and right children are labeled by (u_1, u_2) and (v_1, v_2), respectively, is labeled by one of $(u_1 u_2 v_1, v_2)$, $(u_1, u_2 v_1 v_2)$, $(u_1 v_1, v_2 u_2)$.
- each leaf node is labeled by some (s_1, s_2) such that $s_1 s_2 \in \{b, c\}^* \cup \{a, c\}^* \cup \{a, b\}^*$.

A decomposition is an *n-decomposition* if each node label (u_1, u_2) satisfies $\Psi(u_1, u_2) \in [-n, n] \times [-n, n]$.

Lemma 3. *If $MIX = L(G)$ for some head grammar $G = (\Sigma, N, P, S)$, then there exists an n such that each $w \in MIX$ has an n-decomposition.*

Lemma 4. *If each $w \in MIX$ has an n-decomposition, then each $w \in MIX$ has a 2-decomposition.*

Main Lemma

Lemma.
$z = a^5\, b^{14}\, a^{19}\, c^{29}\, b^{15}\, a^5$ has no 2-decomposition.