MIX Is Not a Tree-Adjoining Language

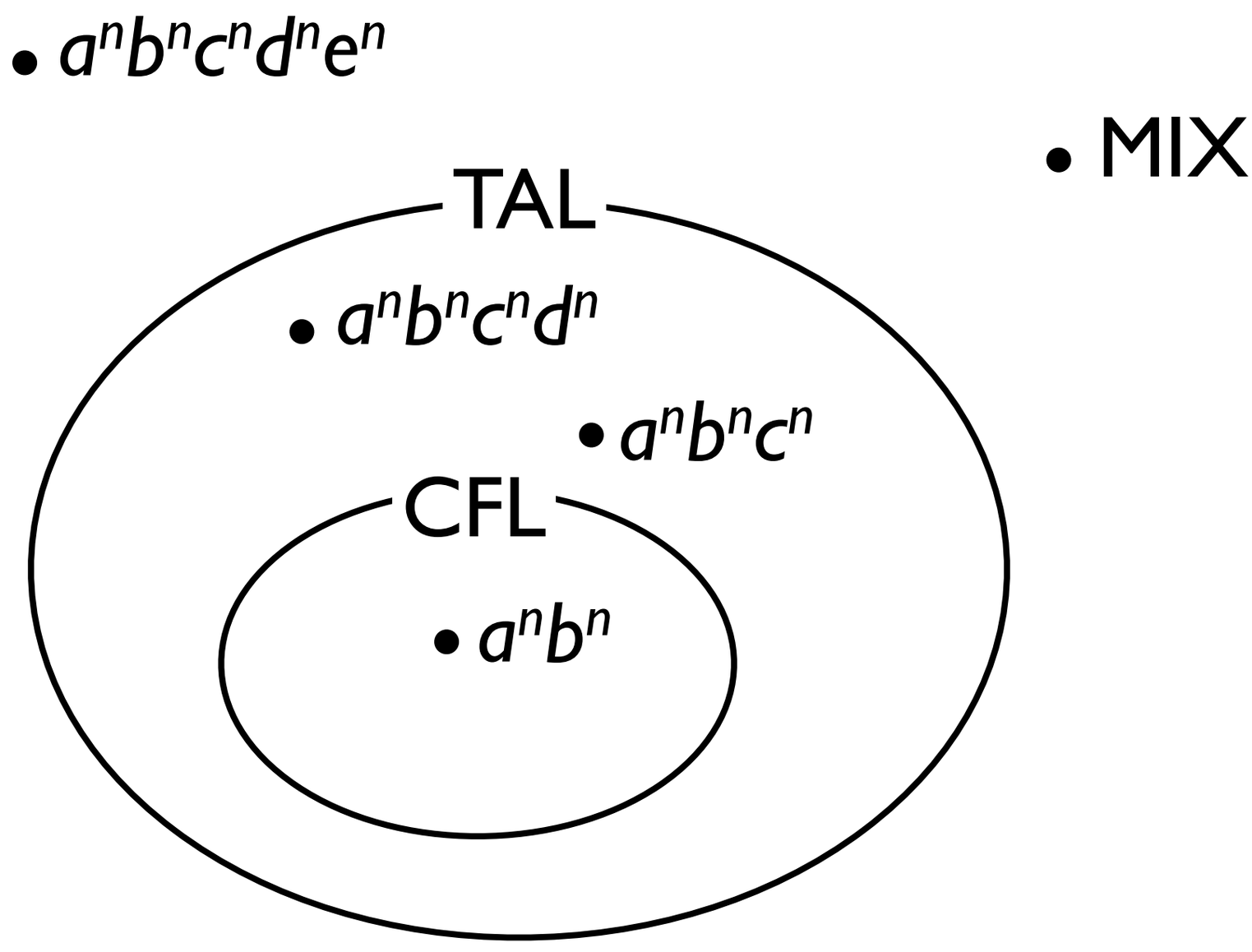
Makoto Kanazawa
NII, Tokyo

Sylvain Salvati
LaBRI/INRIA Bordeaux

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

This talk is about one little language called MIX.
It consists of all strings over {a,b,c} that contain the same number of occurrences of each letter.

MIX $\notin$ TAL

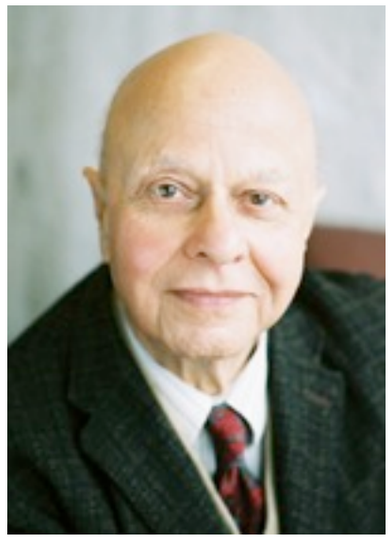


We prove that MIX is not a tree-adjoining language.

Joshi's Conjecture

"TAGs cannot generate this language, although for TAGs the proof is not in hand yet."

Joshi 1985



The non-membership of MIX in TAL was conjectured by Joshi almost 30 years ago. He was confident at the time.

"It is not known whether TAG … can generate MIX. This has turned out to be a very difficult problem."

Joshi, Vijay-Shanker, and Weir 1991



Six years later, the confidence seemed to have faded.

$$\text{MIX} \cap a^*b^*c^* = \{\, a^n b^n c^n \mid n \geq 0 \,\} \notin \text{CFL}$$

Pumping Lemma

$$\text{MIX} \cap R = \quad \notin \text{TAL}$$

Pumping Lemma for TAL

- $a^n b^n c^n d^n e^n$

- MIX

TAL

- $a^n b^n c^n d^n$

Of course, the interest in the question comes from the fact that it was not obvious how to prove it.
The pumping lemma for TAL almost certainly cannot be used to prove that MIX is not in TAL.

MIX and Free Word Order

"[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

A couple of respects in which MIX is not like a natural language.
1. Completely free word order.

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

$$\mathrm{MIX}_k = \{\, w \in \{a_1,\ldots,a_k\}^* \mid |w|_{a1} = \cdots = |w|_{ak} \,\}$$

$$\forall k(\mathrm{MIX}_k \in \mathrm{TAL}) \Rightarrow \forall L(L \in \mathrm{TAL} \Rightarrow \mathrm{Pem}(L) \in \mathrm{TAL})$$

permutation closure

Generalization of MIX.
For TAL, the premise is of course false, but any family of languages that is a “rational cone” or “full trio” and is included in the class of semilinear languages has this property.

MIX Is Not Mildly CS?

"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



2. Exhibits "unlimited (?) cross-serial dependencies".
Joshi et al. 1991 suggested MIX should be excluded from the class of mildly context-sensitive languages because of this property.

Mildly Context-Sensitive Languages

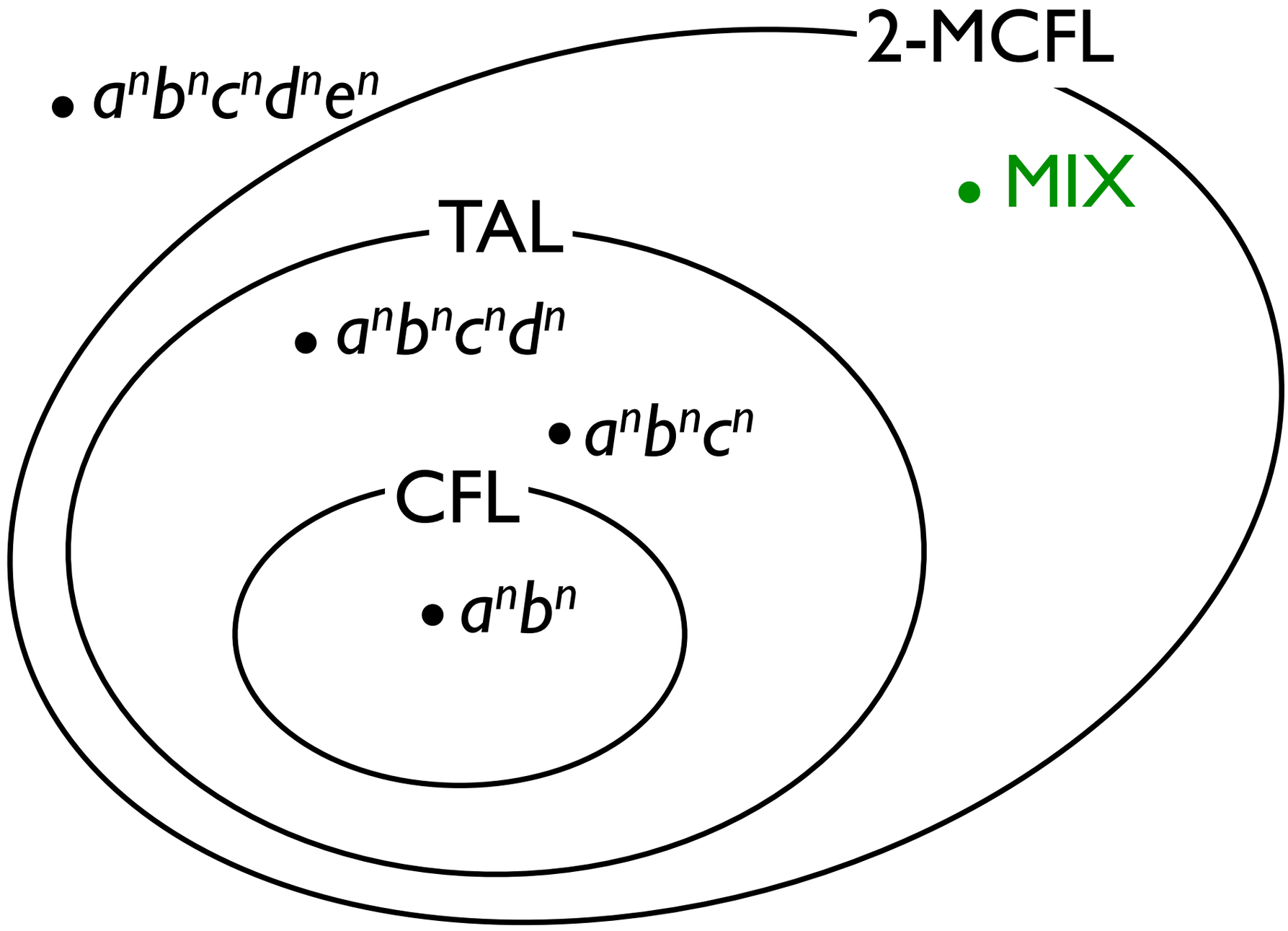
- Polynomial-time recognition
- Semilinearity/constant growth
- Exhibits limited cross-serial dependencies

Joshi 1985

Three defining conditions of mild context-sensitivity.

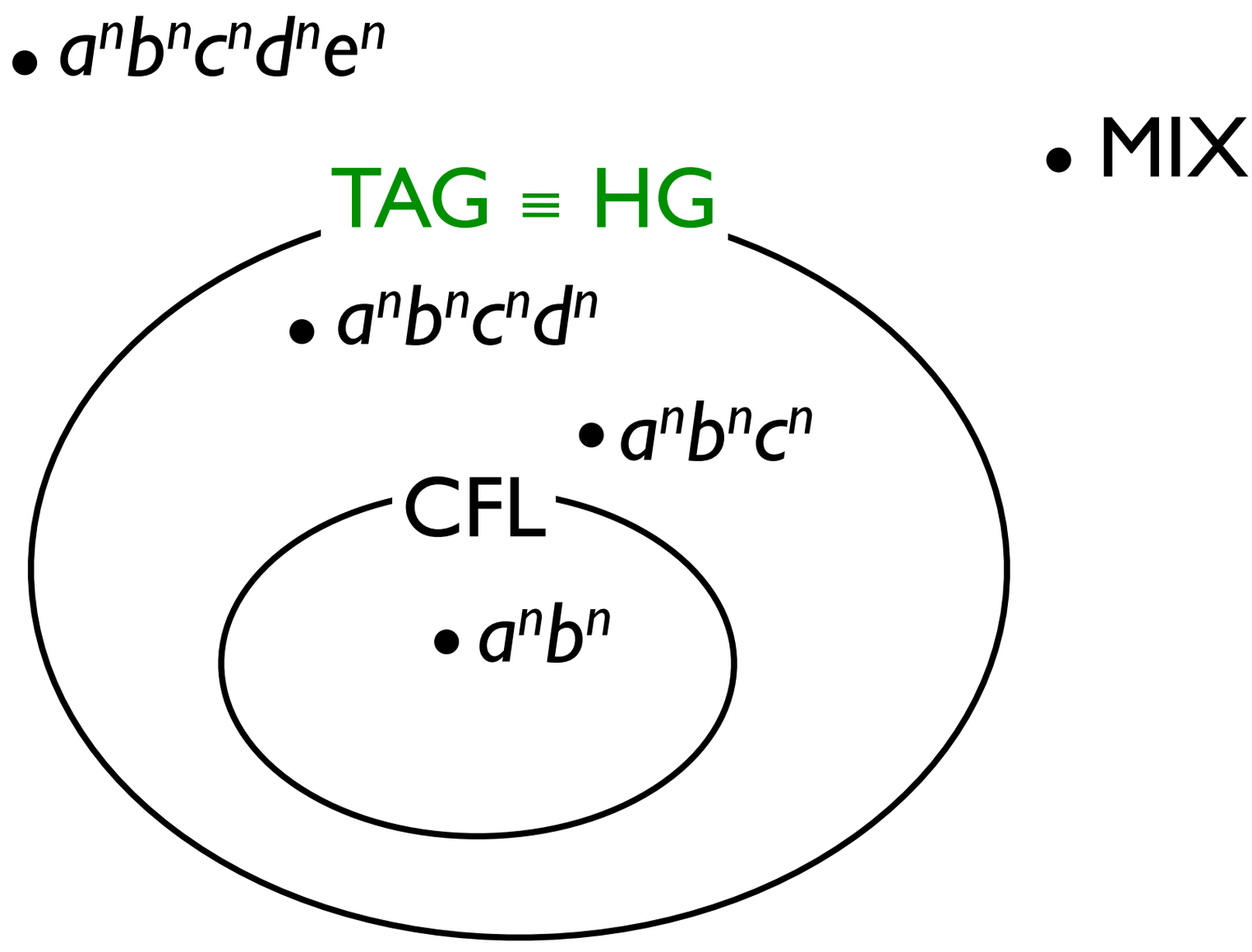
Sylvain Salvati 2011

MIX ∈ 2-MCFL



Surprisingly, MIX is the language of a 2-multiple context-free grammar or linear context-free rewriting system of fanout 2.
Very difficult proof using algebraic topology.

MIX ∉ TAL



Vijay-Shanker and Weir 1994

We use the formalism of head grammar, known to be equivalent to TAG.

Head Grammar

Pollard 1984

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

$A(w_1, w_2) \leftarrow \qquad (w_i \in \Sigma \cup \{\varepsilon\})$

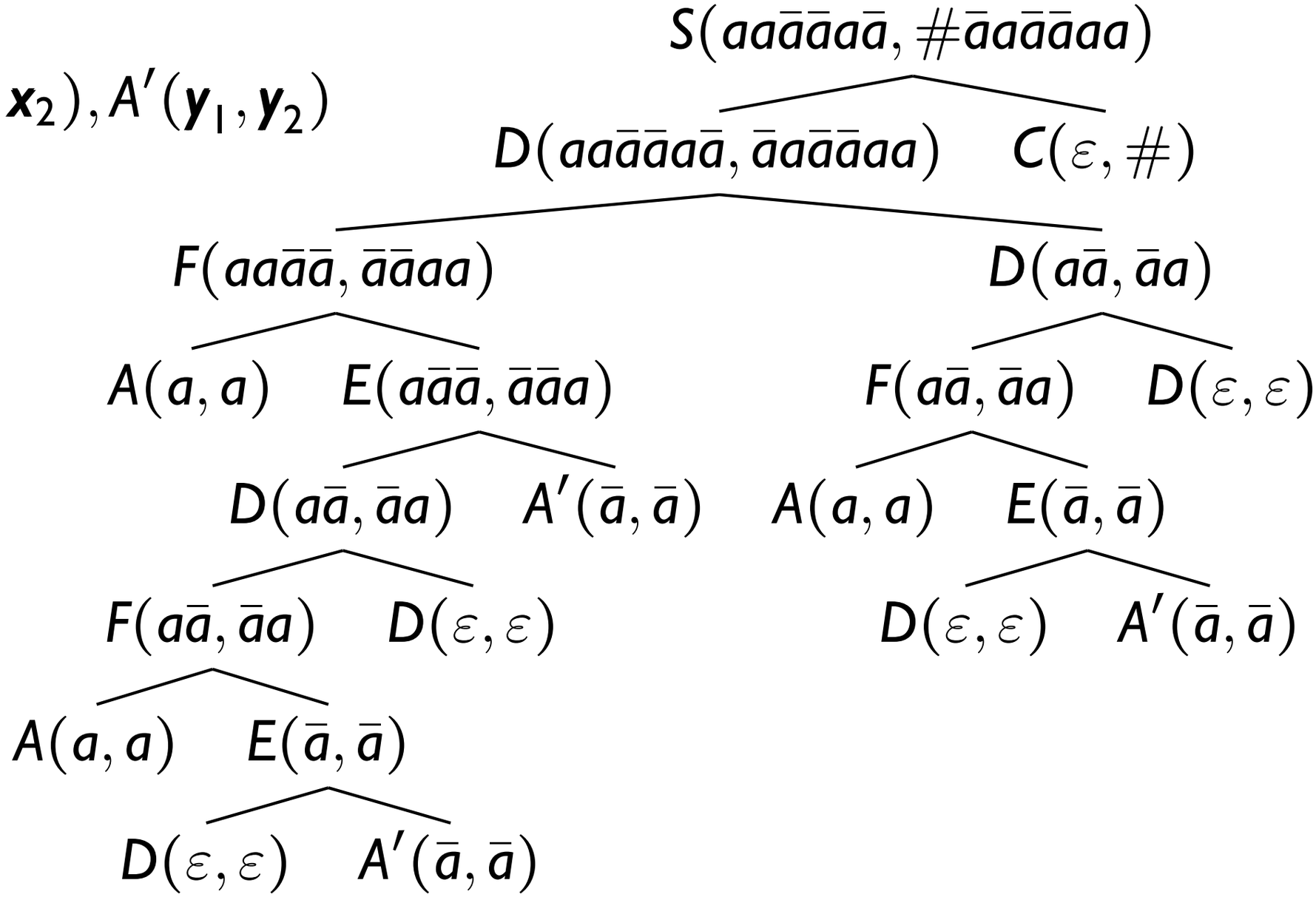
$$L(G) = \{ w_1 w_2 \mid \vdash_G S(w_1, w_2) \}$$

special kind of 2-MCFG or LCFRS(2)

Introduced by Pollard, modified by Roach.
Nonterminals stand for binary relations on strings.
A grammar is a kind of logic program on strings.
Just three operations on pairs of strings.

$$S(x_1 y_1, y_2 x_2) \leftarrow D(x_1, x_2), C(y_1, y_2)$$
$$C(\varepsilon, \#) \leftarrow$$
$$D(\varepsilon, \varepsilon) \leftarrow$$
$$D(x_1 y_1, y_2 x_2) \leftarrow F(x_1, x_2), D(y_1, y_2)$$
$$F(x_1 y_1, y_2 x_2) \leftarrow A(x_1, x_2), E(y_1, y_2)$$
$$A(a, a) \leftarrow$$
$$E(x_1 y_1, y_2 x_2) \leftarrow D(x_1, x_2), A'(y_1, y_2)$$
$$A'(\bar{a}, \bar{a}) \leftarrow$$

$$\{ w\#w^R \mid w \in D_1 \}$$



An example of a head grammar (kind of synchronous CFG).
All non-terminating rules are wrapping rules.

$S(x_1y_1, y_2x_2) \leftarrow D(x_1, x_2), C(y_1, y_2)$

$C(\varepsilon, \#) \leftarrow$

$D(\varepsilon, \varepsilon) \leftarrow$

$D(x_1y_1, y_2x_2) \leftarrow F(x_1, x_2), D(y_1, y_2)$

$F(x_1y_1, y_2x_2) \leftarrow A(x_1, x_2), E(y_1, y_2)$

$A(a, a) \leftarrow$

$E(x_1y_1, y_2x_2) \leftarrow D(x_1, x_2), A'(y_1, y_2)$

$A'(\bar{a}, \bar{a}) \leftarrow$

$\{ w\#w^R \mid w \in D_1 \}$

HG ≡ well-nested 2-MCFG

$S(x_1y_1, x_2y_2) \leftarrow D(x_1, x_2), C(y_1, y_2)$

$C(\varepsilon, \#) \leftarrow$

$D(\varepsilon, \varepsilon) \leftarrow$

$D(x_1y_1, x_2y_2) \leftarrow F(x_1, x_2), D(y_1, y_2)$

$F(x_1y_1, x_2y_2) \leftarrow A(x_1, x_2), E(y_1, y_2)$

$A(a, a) \leftarrow$

$E(x_1y_1, x_2y_2) \leftarrow D(x_1, x_2), A'(y_1, y_2)$

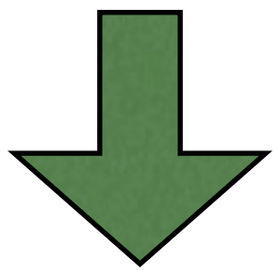
$A'(\bar{a}, \bar{a}) \leftarrow$

$\{ w\#w \mid w \in D_1 \}$

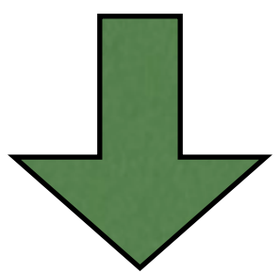
non-well-nested 2-MCFG

String operations used in HG are “well-nested”.

$\exists G \in \mathrm{HG}(L(G) = \mathrm{MIX})$



$\exists n \forall w \in \mathrm{MIX}(w \text{ has } n\text{-decomposition})$

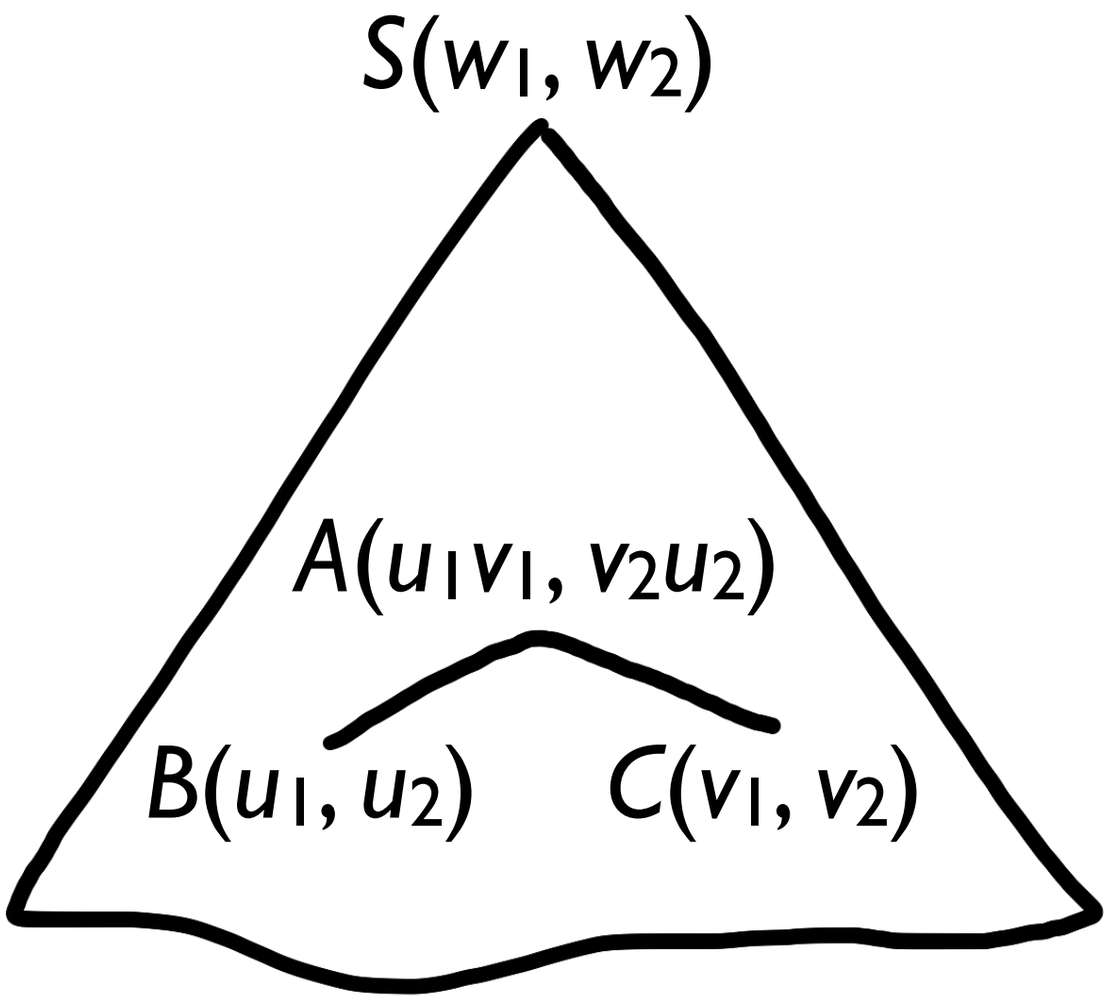


$\forall w \in \mathrm{MIX}(w \text{ has 2-decomposition})$

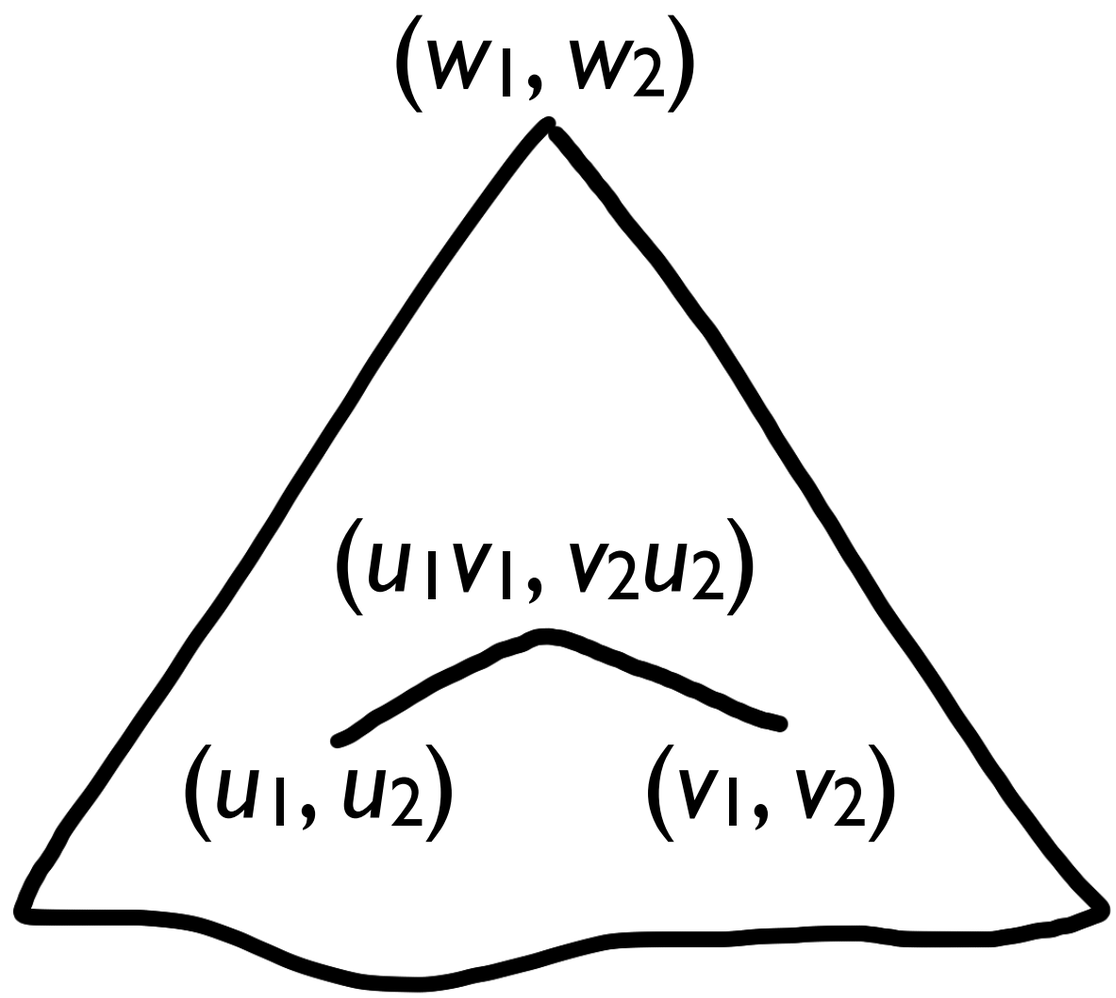
$a^5b^{14}a^{19}c^{29}b^{15}a^5$ has no 2-decomposition.

Outline of the proof

derivation of $w_1 w_2$

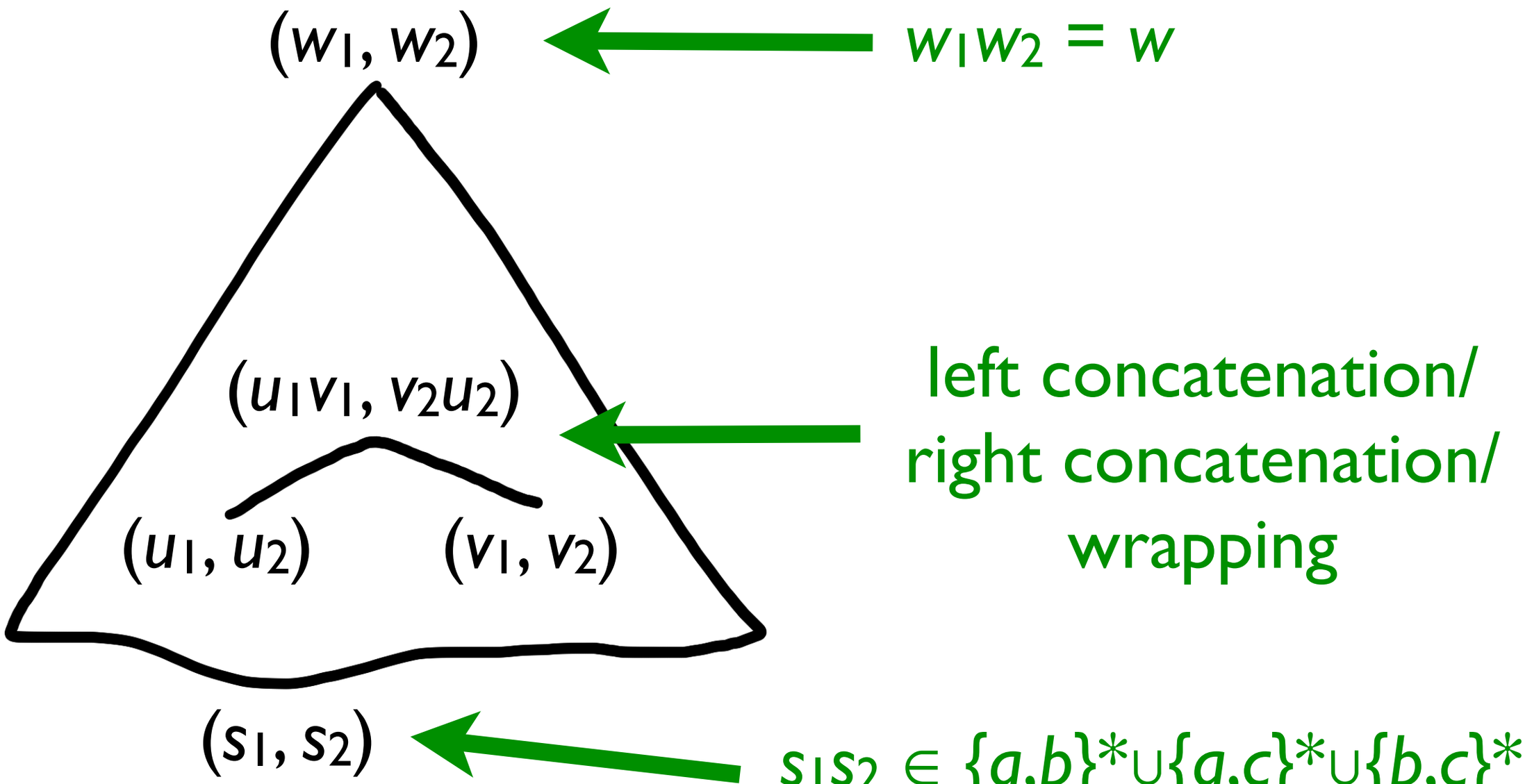


decomposition of $w_1 w_2$



A decomposition is similar to an HG derivation but independent of any grammar.
Any HG derivation can be turned into a decomposition by stripping off nonterminals.

decomposition of w



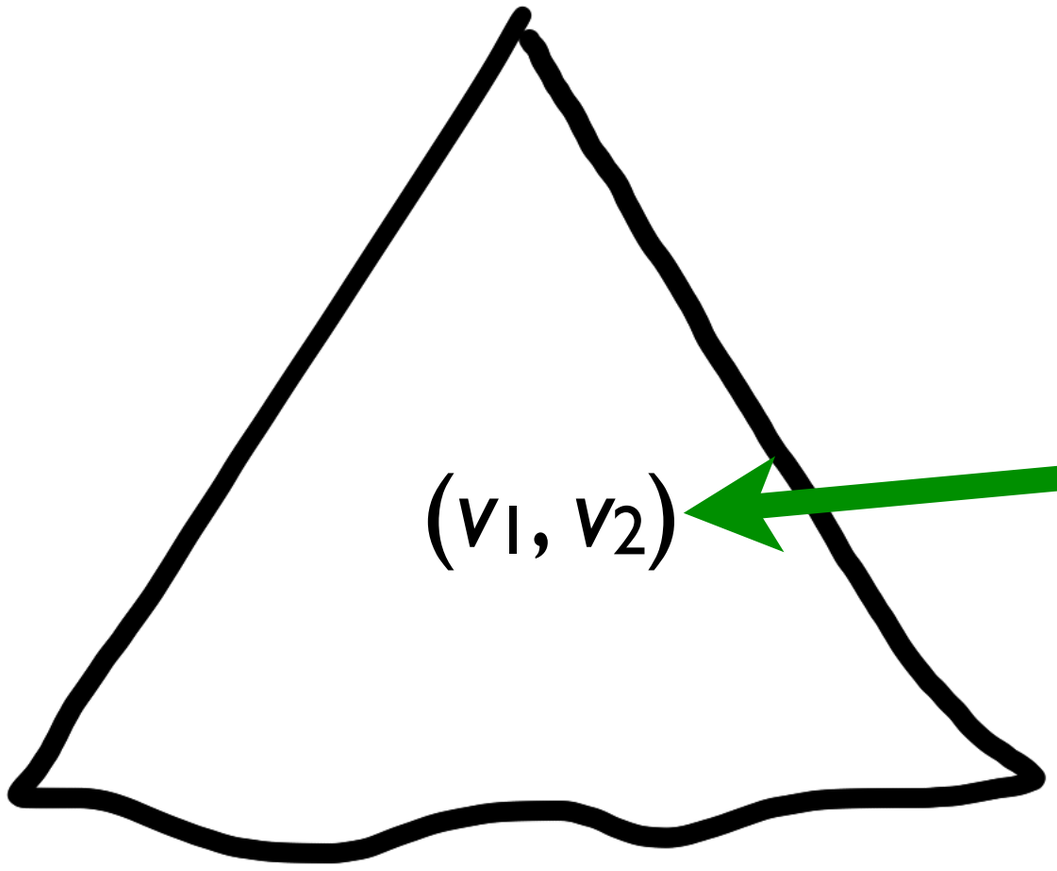
Formally, a decomposition is a binary tree whose nodes are labeled by pairs of strings.
Every HG derivation of w gives a decomposition of w.

n-decomposition of *w*

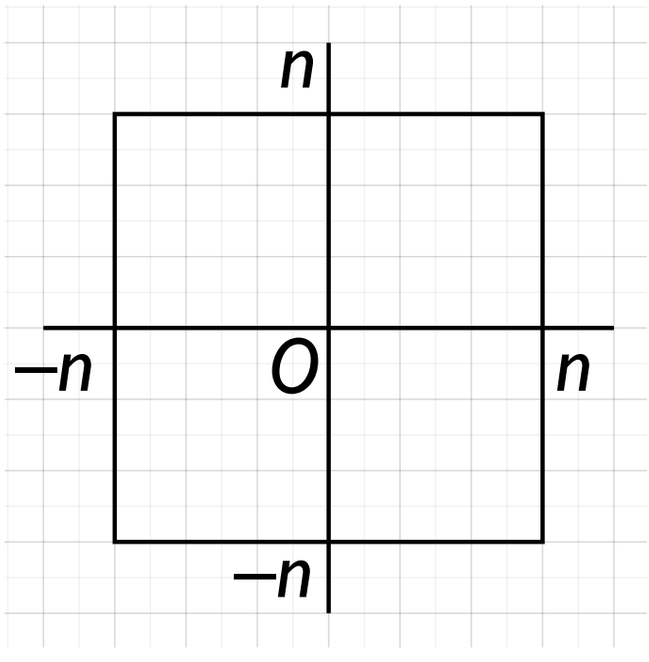
$$\Psi_1(v) = |v|_a - |v|_c$$

$$\Psi_2(v) = |v|_b - |v|_c$$

$$\Psi(v) = (\Psi_1(v), \Psi_2(v))$$



$\Psi(v_1 v_2) \in [-n, n] \times [-n, n]$



The parameter n measures how unbalanced the occurrence counts of the three letters can be at any node in a decomposition.

$$\Psi_1(v) = |v|_a - |v|_c$$

$$\Psi_2(v) = |v|_b - |v|_c$$

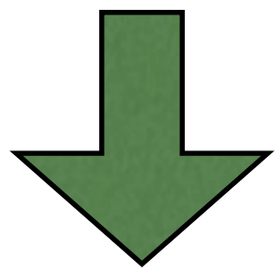
$$\Psi(v) = (\Psi_1(v), \Psi_2(v))$$

$$w \in \mathrm{MIX} \Leftrightarrow \Psi(w) = (0, 0)$$

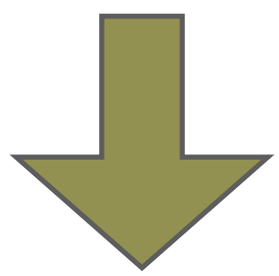
$$u_0 v_1 u_1 v_2 u_2, u_0 v_1' u_1 v_2' u_2 \in \mathrm{MIX} \Rightarrow \Psi(v_1 v_2) = \Psi(v_1' v_2')$$

Properties of the function ψ

$\exists G \in \mathrm{HG}(L(G) = \mathrm{MIX})$



$\exists n \forall w \in \mathrm{MIX}(w \text{ has } n\text{-decomposition})$

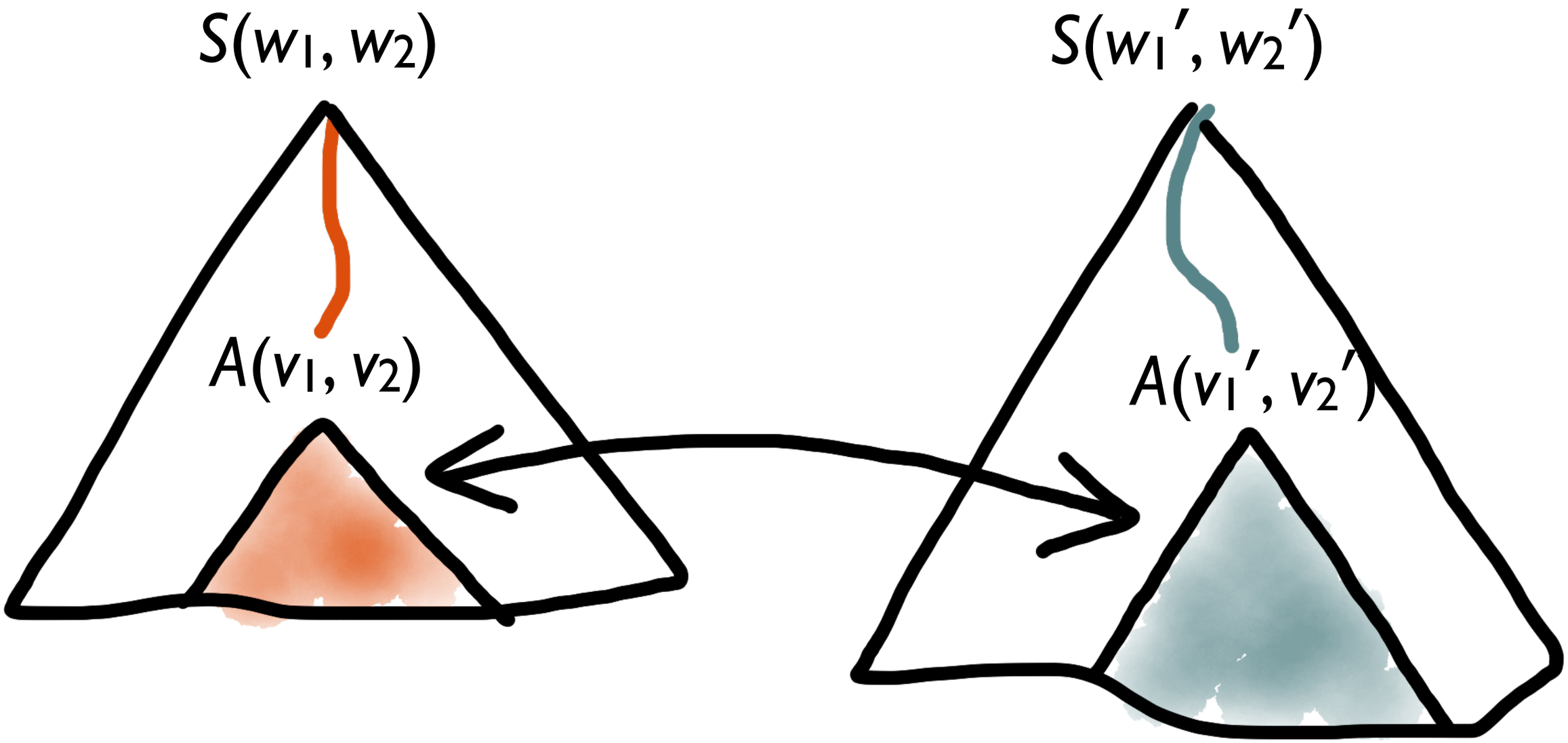


$\forall w \in \mathrm{MIX}(w \text{ has 2-decomposition})$

$a^5b^{14}a^{19}c^{29}b^{15}a^5$ has no 2-decomposition.

Prove the first implication.

Suppose $L(G)$ = MIX.

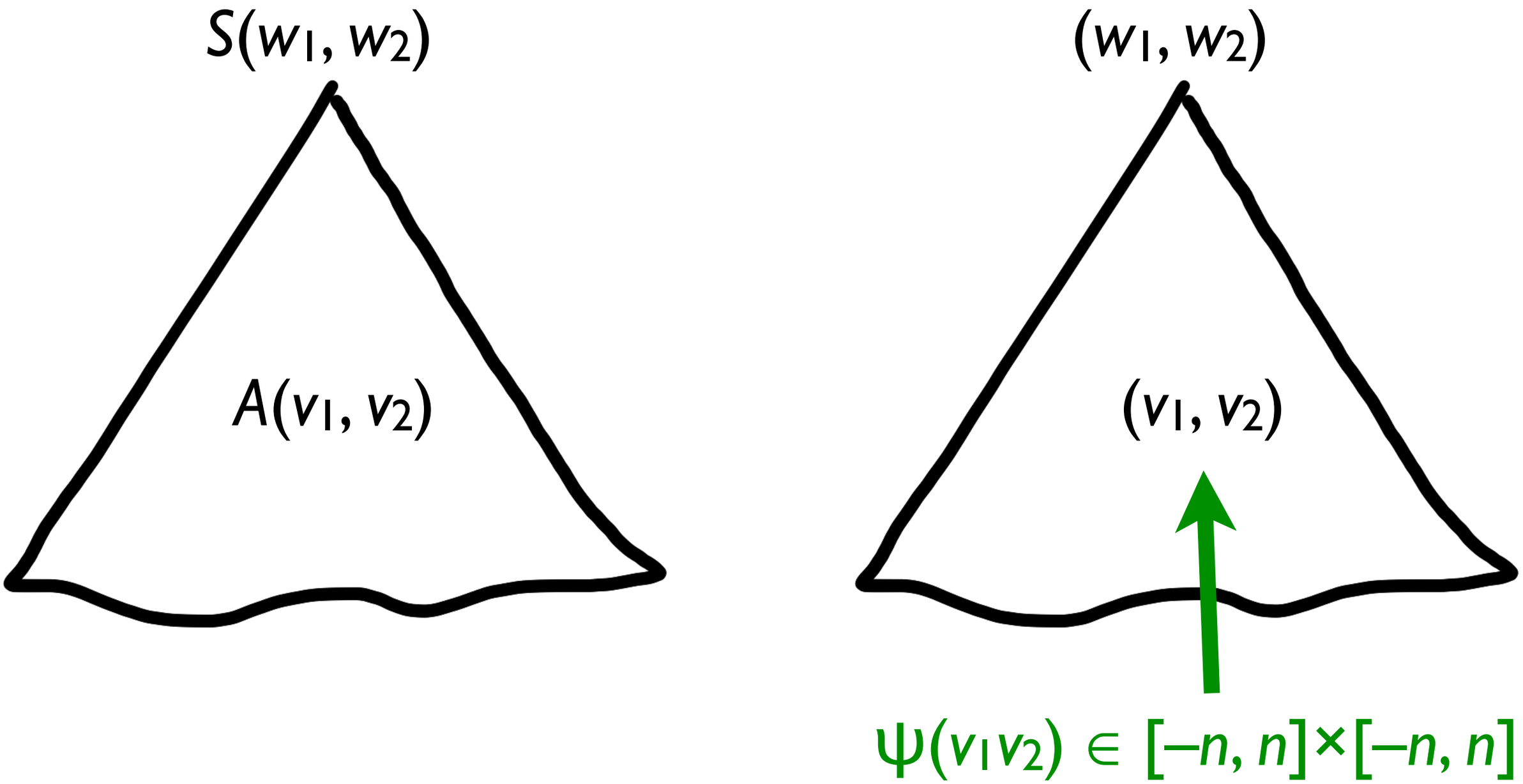


$$\vdash_G A(v_1, v_2), \vdash_G A(v_1', v_2') \Rightarrow \Psi(v_1 v_2) = \Psi(v_1' v_2')$$

Whenever the same nonterminal holds of two pairs of strings, their ψ value is the same.

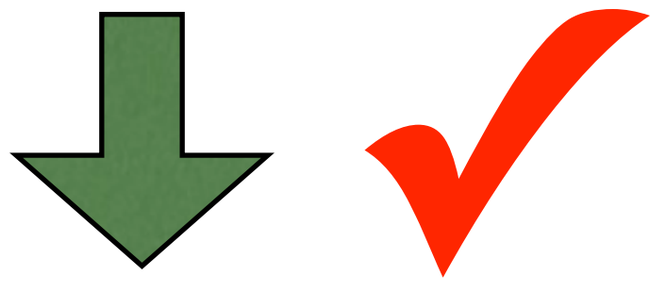
Let $n = \max\{ \|\psi(v_1 v_2)\|_\infty \mid \vdash_G A(v_1, v_2) \text{ for some } A \}$.

$\{ \psi(v_1 v_2) \mid \vdash_G A(v_1, v_2) \text{ for some } A \} \subseteq [-n, n] \times [-n, n]$

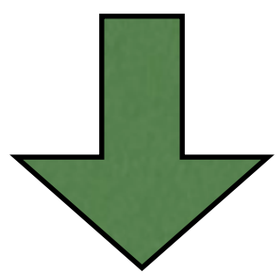


A head grammar G determines the value of n.
Every decomposition that comes from a derivation according to G is an n-decomposition.

$\exists G \in \mathrm{HG}(L(G) = \mathrm{MIX})$



$\exists n \forall w \in \mathrm{MIX}(w \text{ has } n\text{-decomposition})$



$\forall w \in \mathrm{MIX}(w \text{ has 2-decomposition})$

$a^5b^{14}a^{19}c^{29}b^{15}a^5$ has no 2-decomposition.

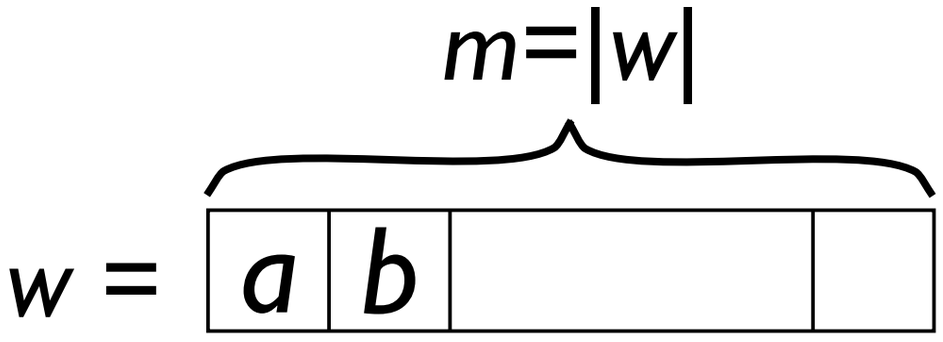
The first implication done.
Now prove the second implication.

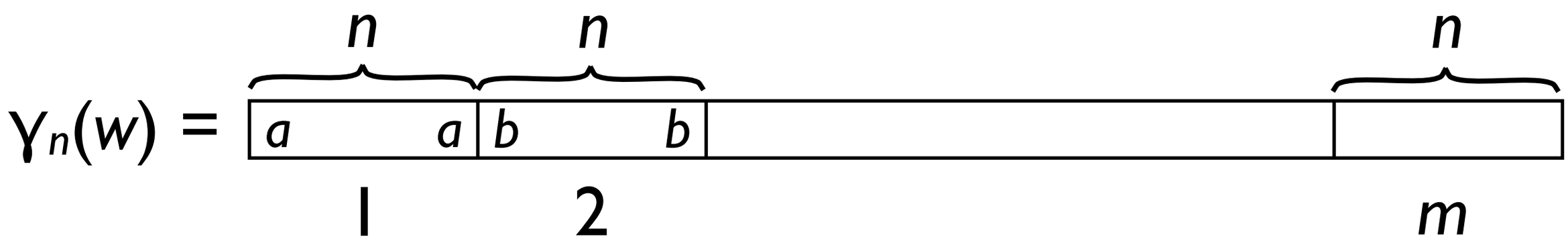
$$\gamma_n(a) = a^n$$

$$\gamma_n(b) = b^n$$

$$\gamma_n(c) = c^n$$

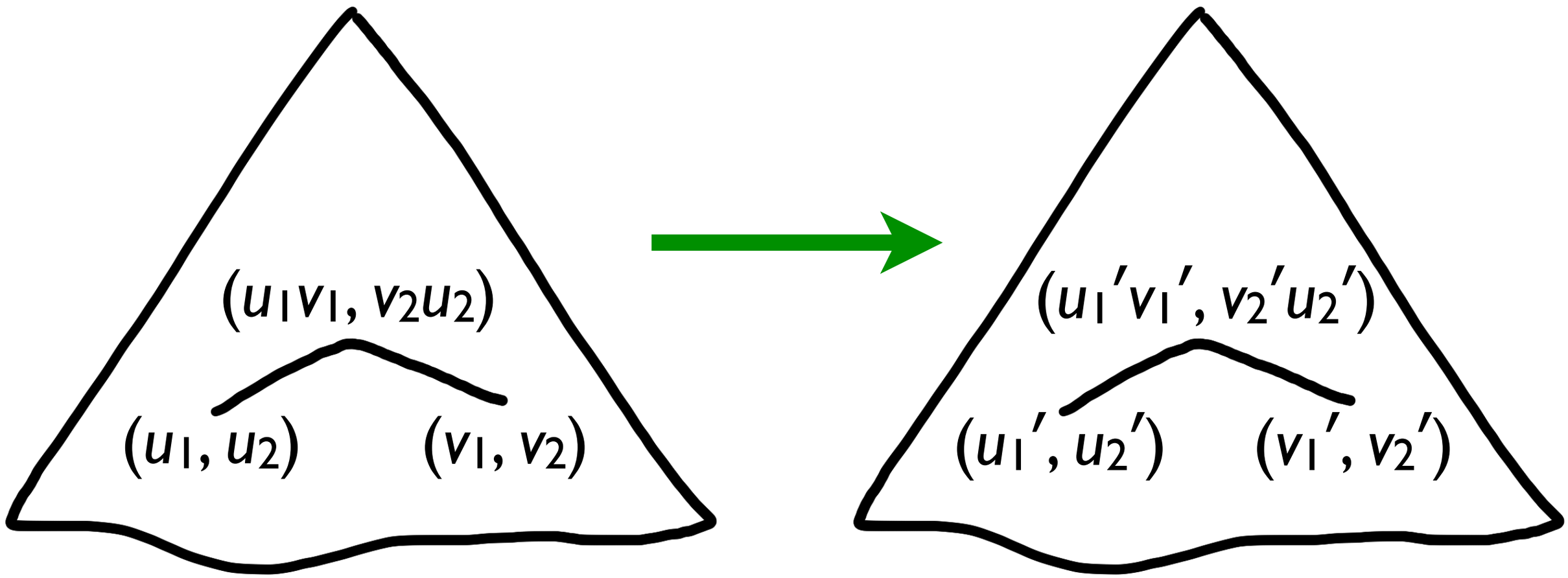
homomorphism

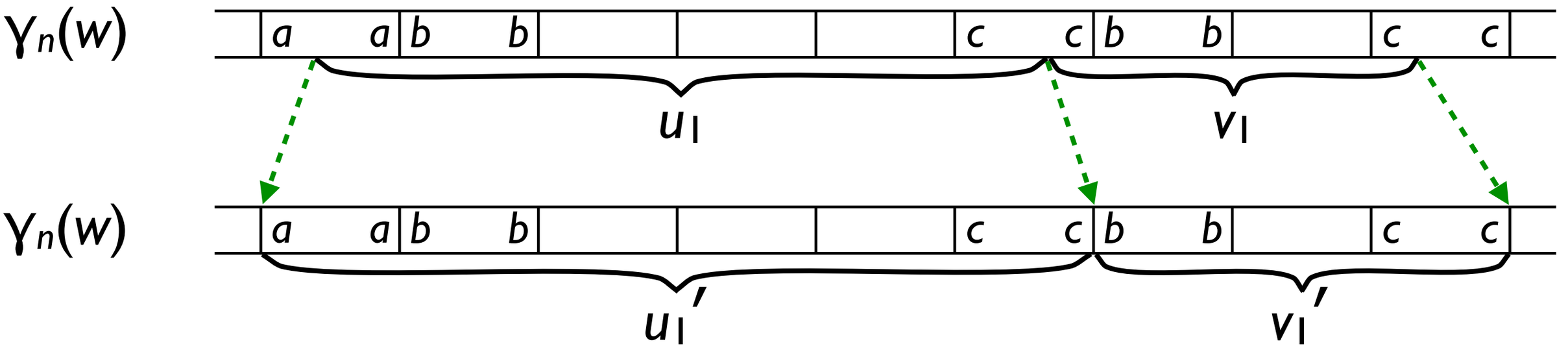




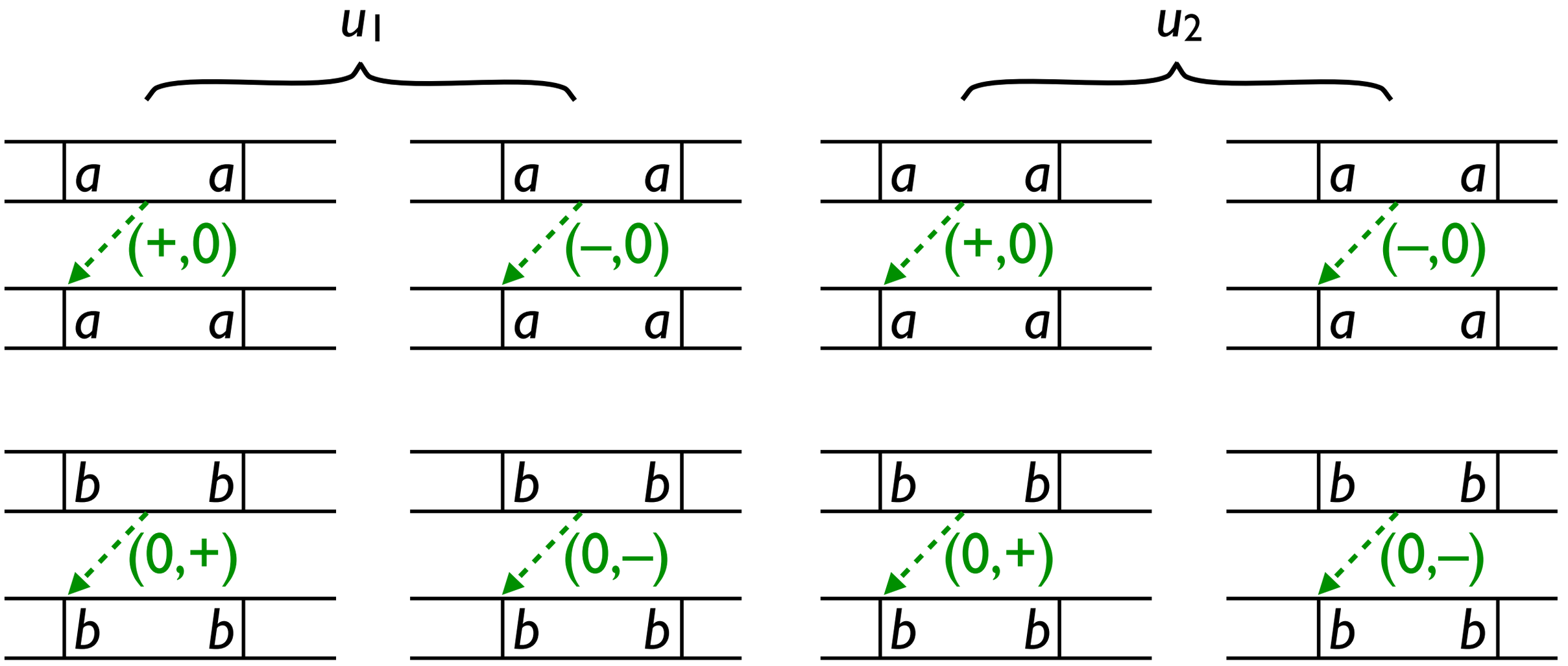
m blocks of size *n*

The proof uses a homomorphism that repeats each letter n times.





The goal is to show that w ∈ MIX has a 2-decomposition.
Start with an n-decomposition of $\gamma_n(w)$.
Turn it into a "neat" decomposition.
Primed strings start and end at block boundaries.



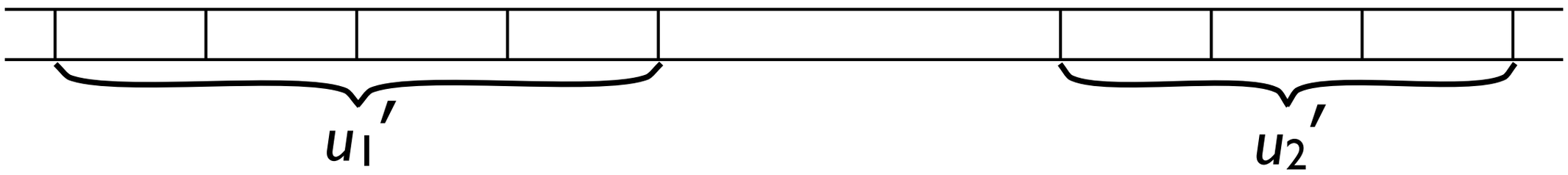
$\Psi_i(u_1') - \Psi_i(u_1) \in [-n+1, n-1]$ $\Psi_i(u_2') - \Psi_i(u_2) \in [-n+1, n-1]$

The “unbalancedness” may increase after the transformation.

$$\psi_i(u_1'u_2')-\psi_i(u_1u_2) \in [-2n+2, 2n-2]$$

$$\psi(u_1u_2) \in [-n, n]\times[-n, n]$$

$$\psi(u_1'u_2') \in [-3n+2, 3n-2]\times [-3n+2, 3n-2]$$

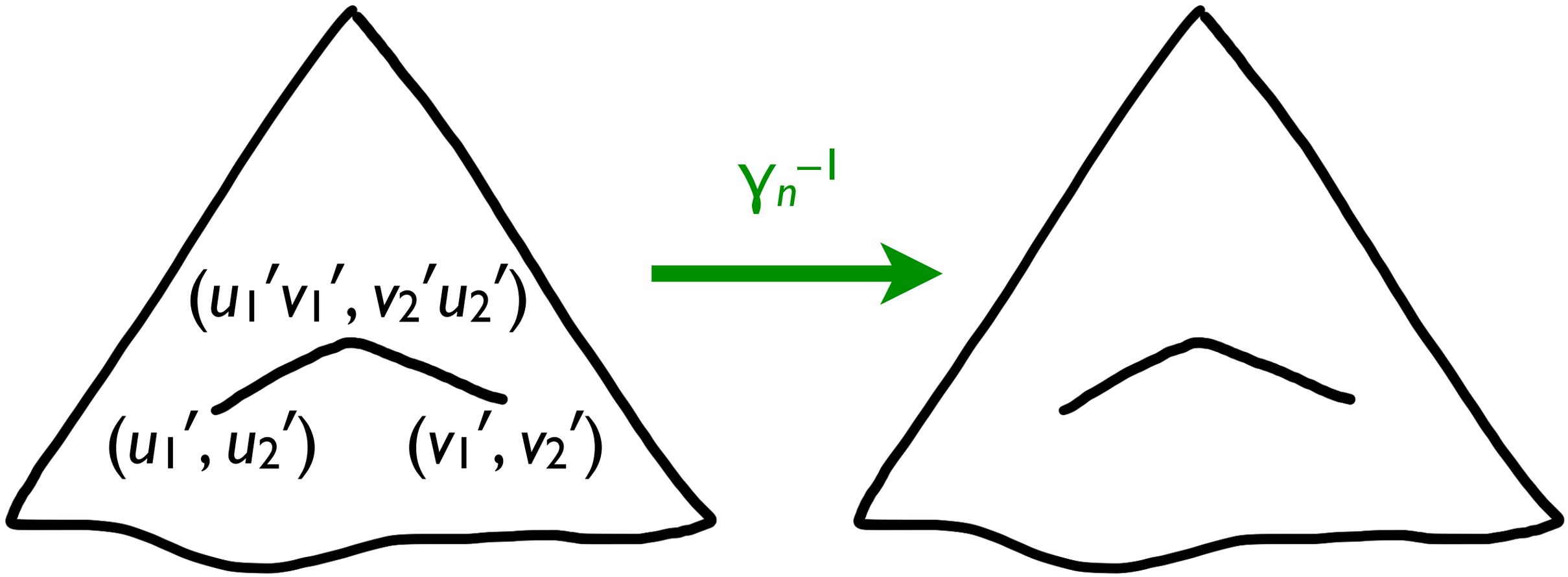


$$\psi(u_1'u_2') = (pn, qn)$$

$$\psi(u_1'u_2') \in [-2n, 2n]\times [-2n, 2n]$$

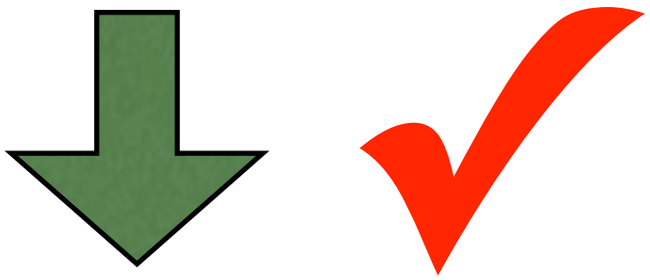
2*n*-decomposition of $\gamma_n(w)$

2-decomposition of *w*

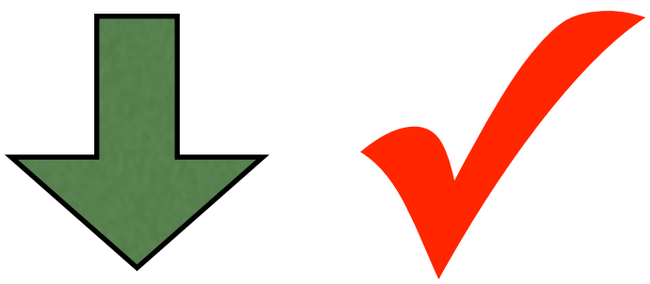


Invert the homomorphism γ_n and obatin a 2-decomposition of w.

$\exists G \in \mathrm{HG}(L(G) = \mathrm{MIX})$



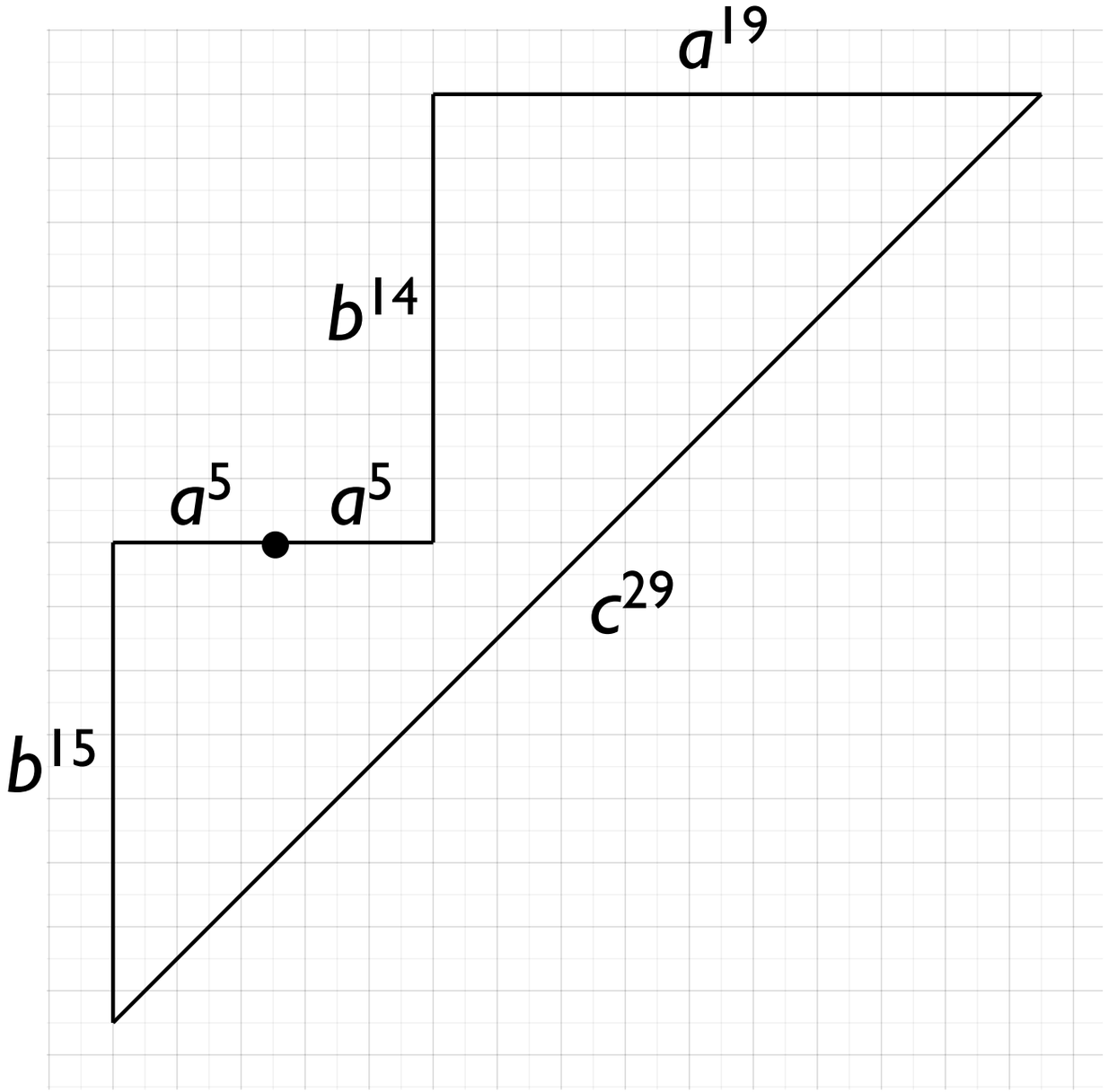
$\exists n \forall w \in \mathrm{MIX}(w \text{ has } n\text{-decomposition})$



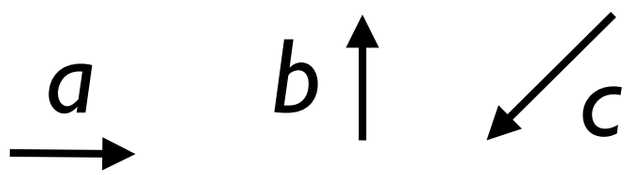
$\forall w \in \mathrm{MIX}(w \text{ has 2-decomposition})$

$a^5b^{14}a^{19}c^{29}b^{15}a^5$ has no 2-decomposition.

$a^5b^{14}a^{19}c^{29}b^{15}a^5$ has no 2-decomposition.

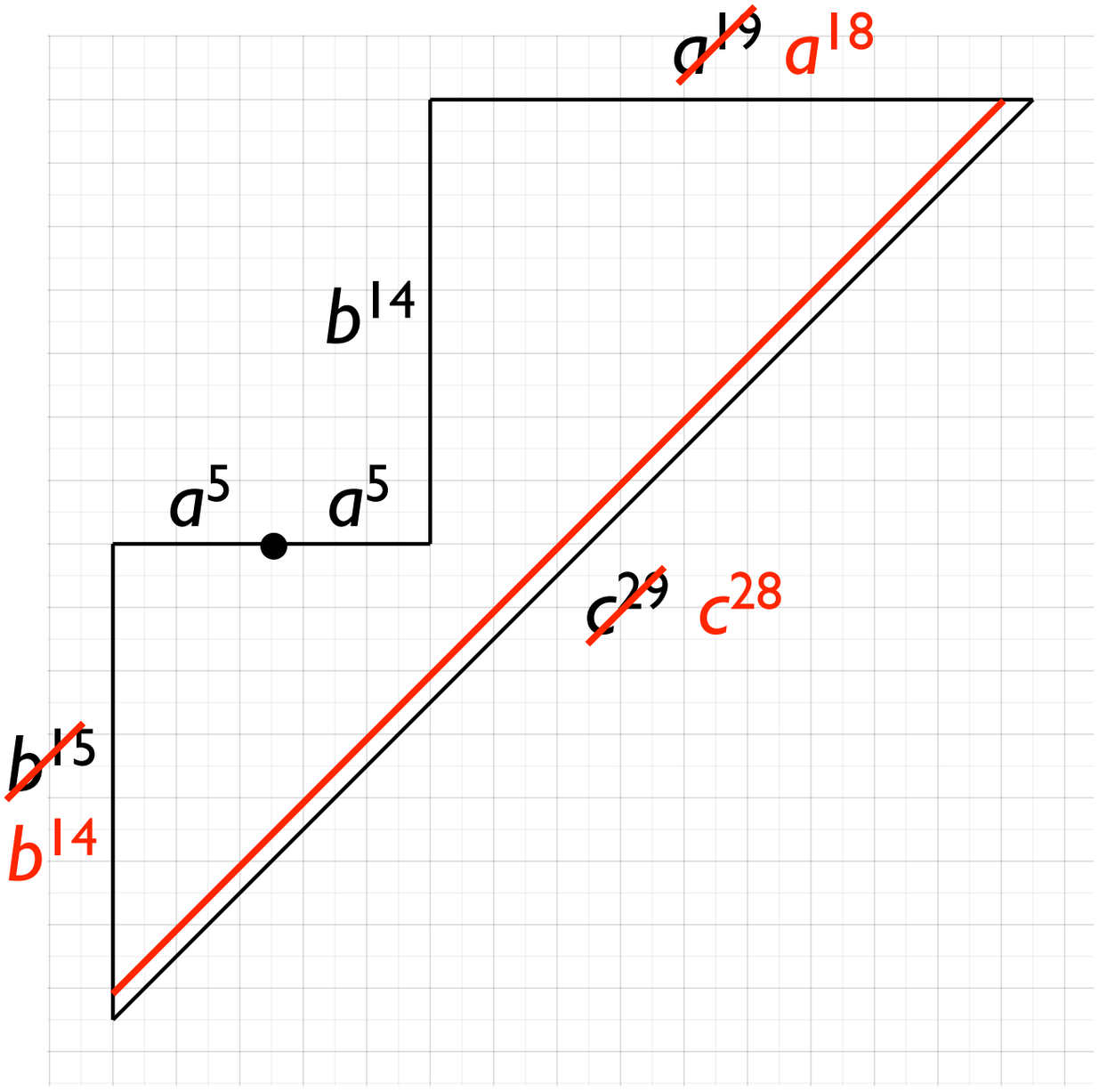


- computer verification
- mathematical proof



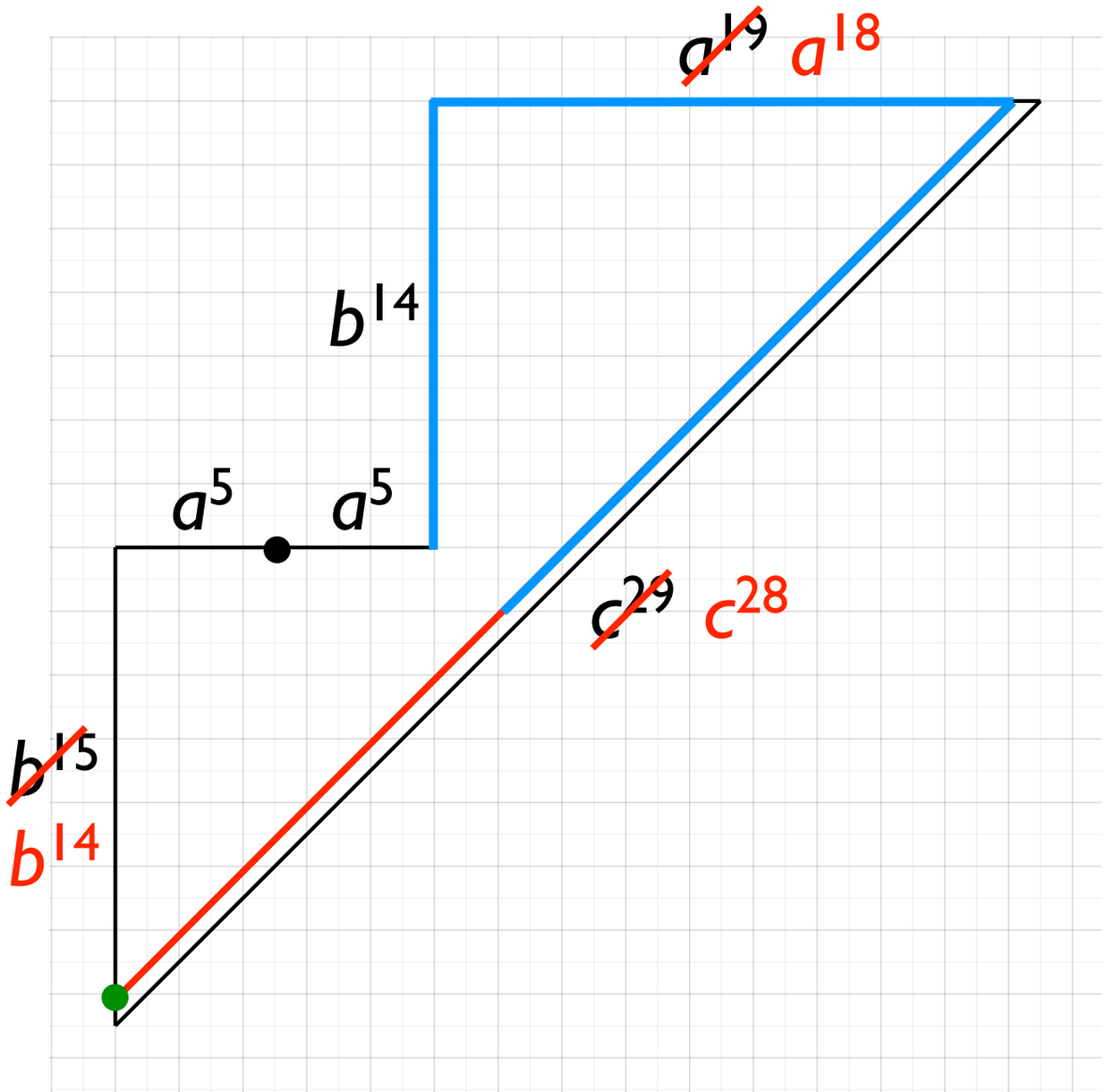
Can show this in two ways.
Computer program is available from my web site.
There's no easy way to explain the mathematical proof.

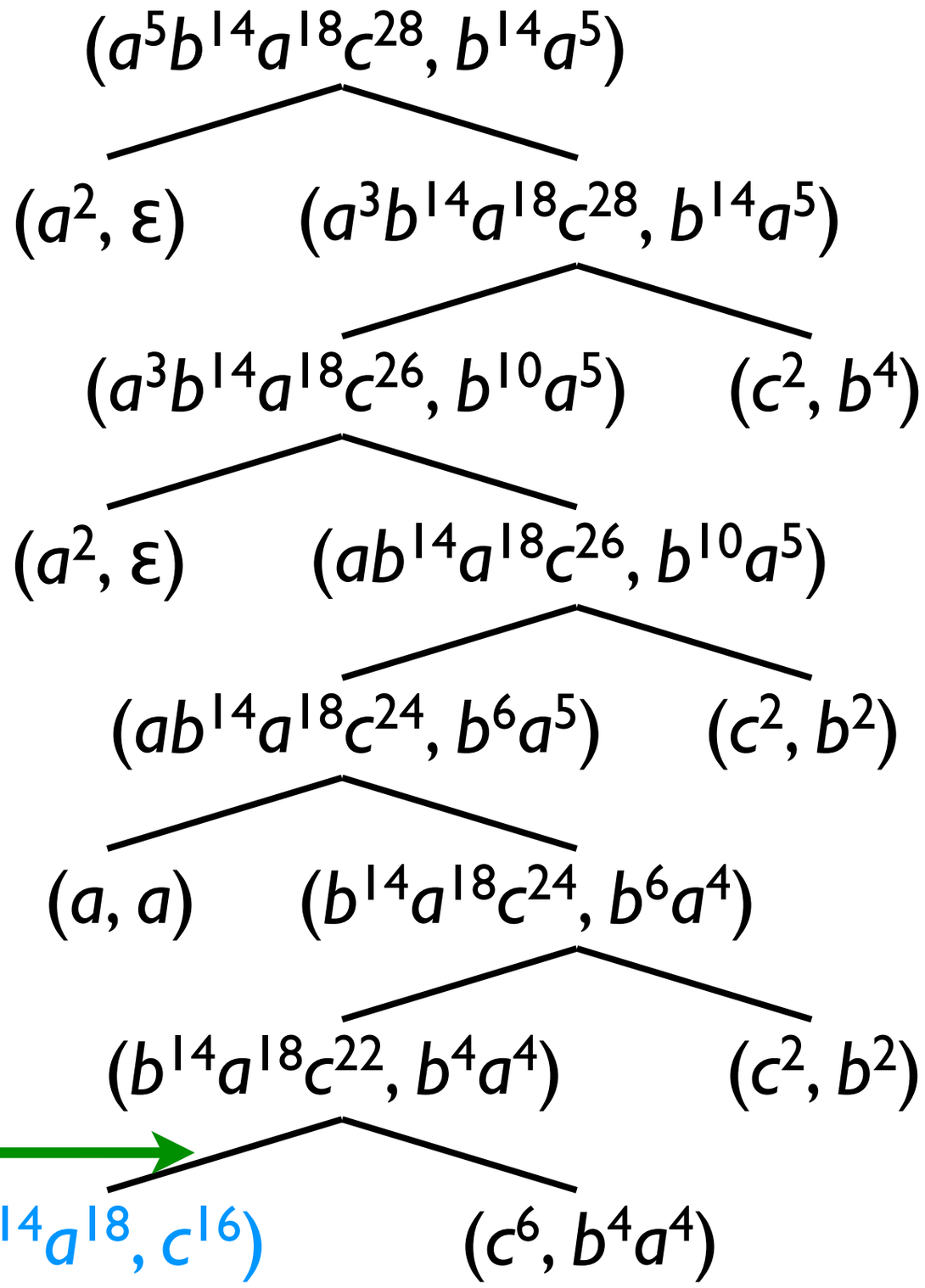
$a^5b^{14}a^{18}c^{28}b^{14}a^5$ has a 2-decomposition.



We believe our counterexample is minimal.
A slight modification leads to a 2-decomposable string.

$a^5b^{14}a^{18}c^{28}b^{14}a^5$ has a 2-decomposition.

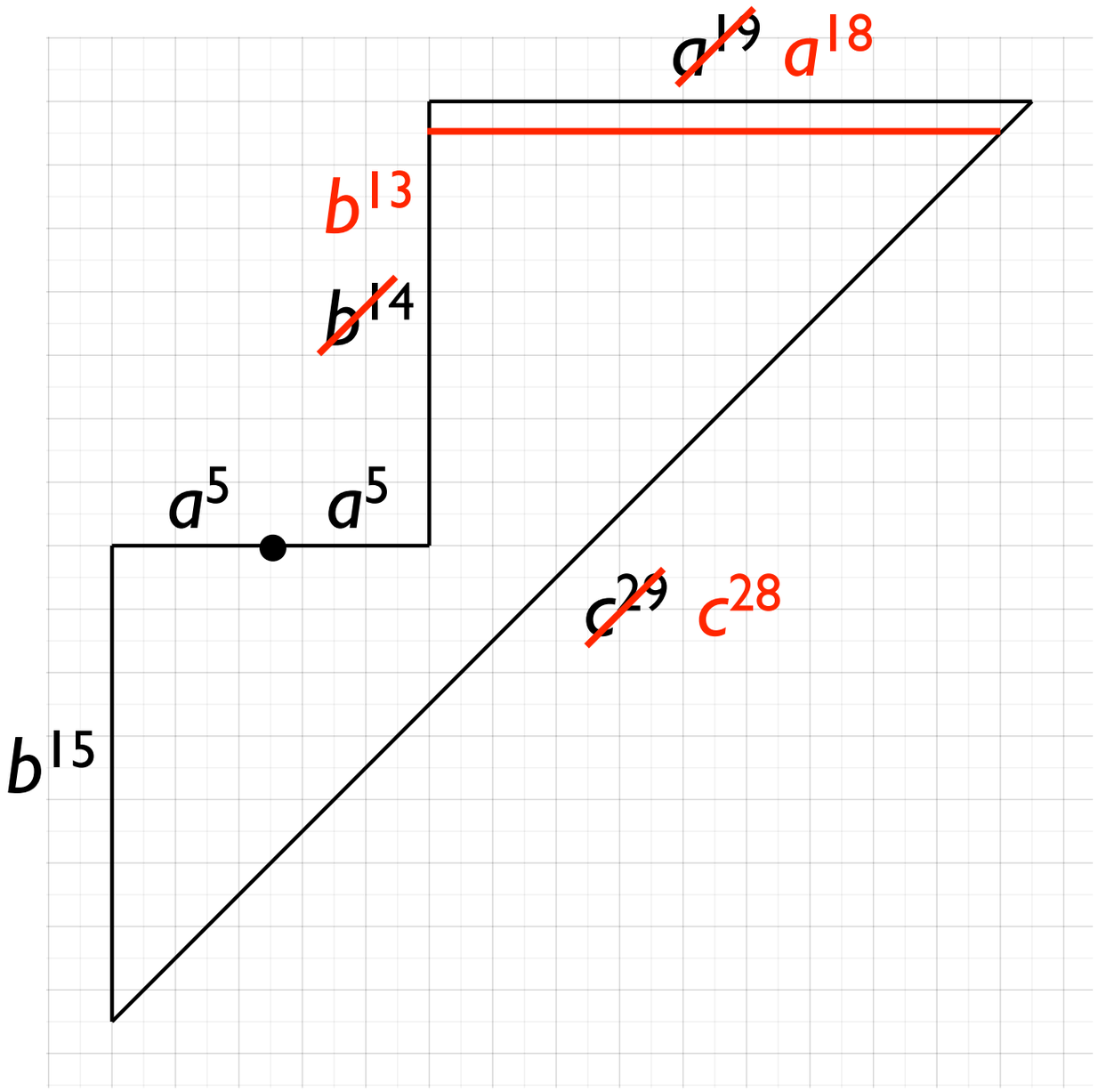




left concatenation →

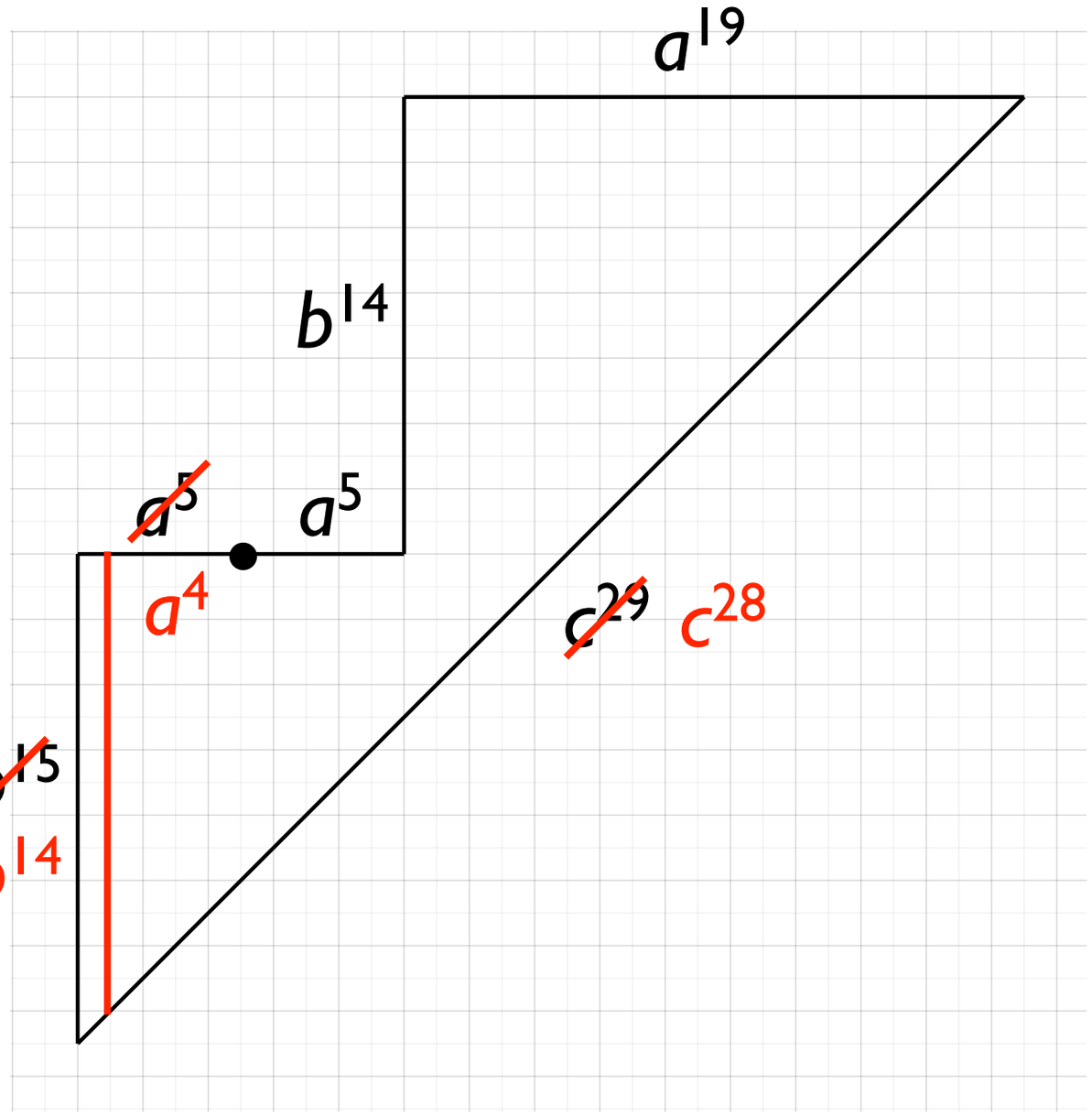
The “slope” too close to the corner to the right of the origin.

$a^5b^{13}a^{18}c^{28}b^{15}a^5$ has a 2-decomposition.



Another simplification.

$a^5b^{14}a^{18}c^{28}b^{14}a^4$ has a 2-decomposition.



Yet another simplification.

2012

Kanazawa and Salvati
MIX is not a tree-adjoining language.

2011

Sylvain Salvati
MIX is a 2-MCFL.

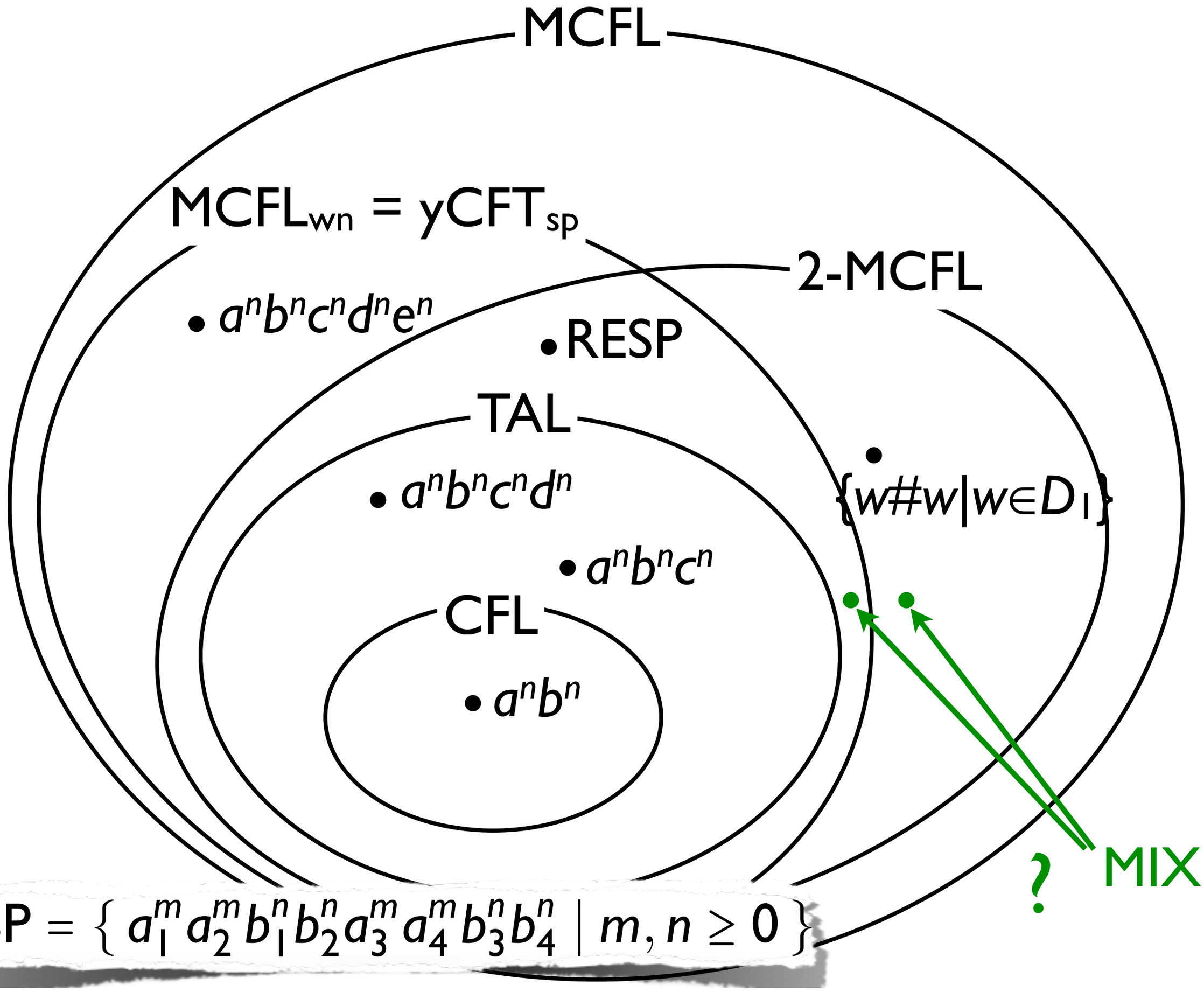
Joshi, Vijay-Shanker, and Weir
This has turned out to be a very difficult problem.

1991

1985

Aravind K. Joshi
TAGs cannot generate this language, although for TAGs the proof is not in hand yet.

Now we know much more about MIX than before.



MIX belongs to 2-MCFL – TAL.
There are two other such languages.
RESP belongs to MCFL$_{wn}$.
What about MIX?