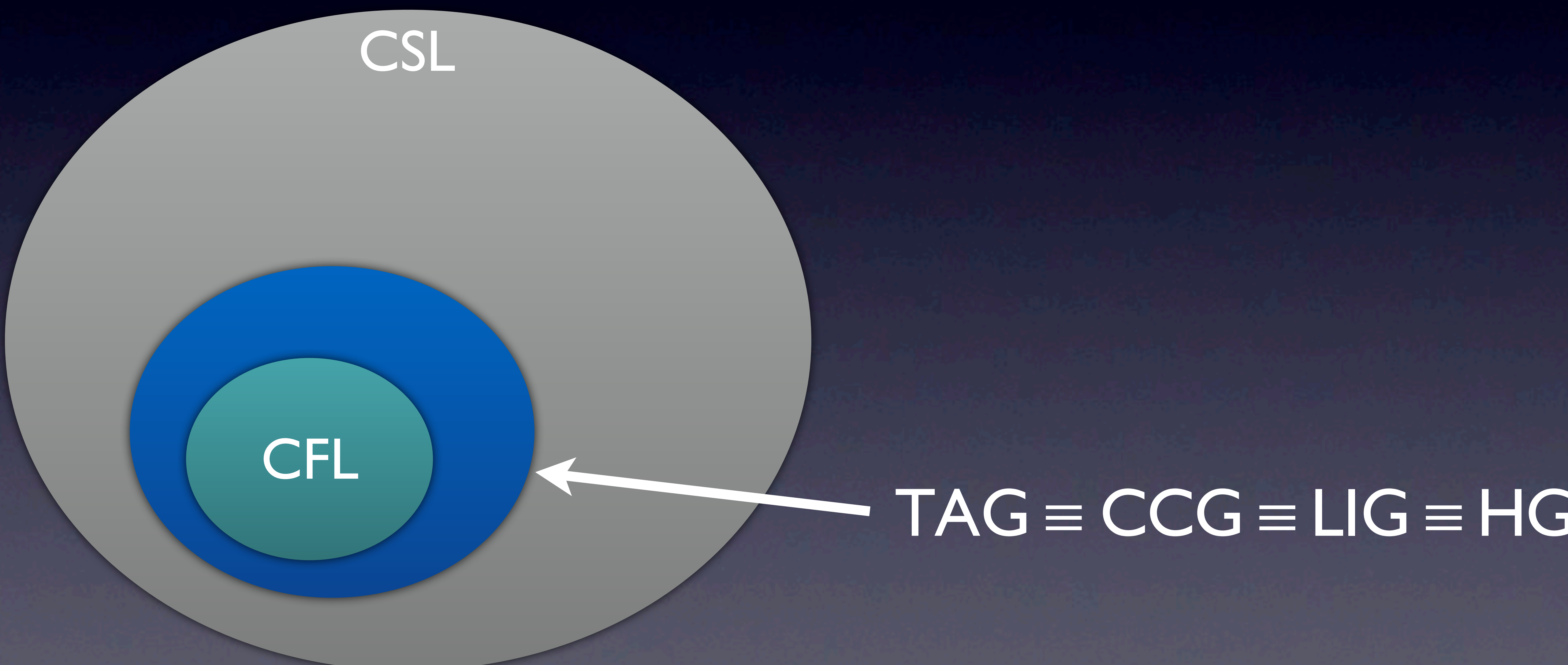
The Convergence of Well-Nested Mildly Context-Sensitive Grammar Formalisms

Makoto Kanazawa
National Institute of Informatics
Tokyo, Japan

The convergence of mildly context-sensitive grammar formalisms



Joshi, Vijay-Shanker, and Weir 1991

Title of this talks comes from Joshi et al.'s paper
Four independently developed formalisms found to be equivalent
The class of languages characterized by the four formalisms is robust, which shows its importance

Mildly context-sensitive grammar formalisms

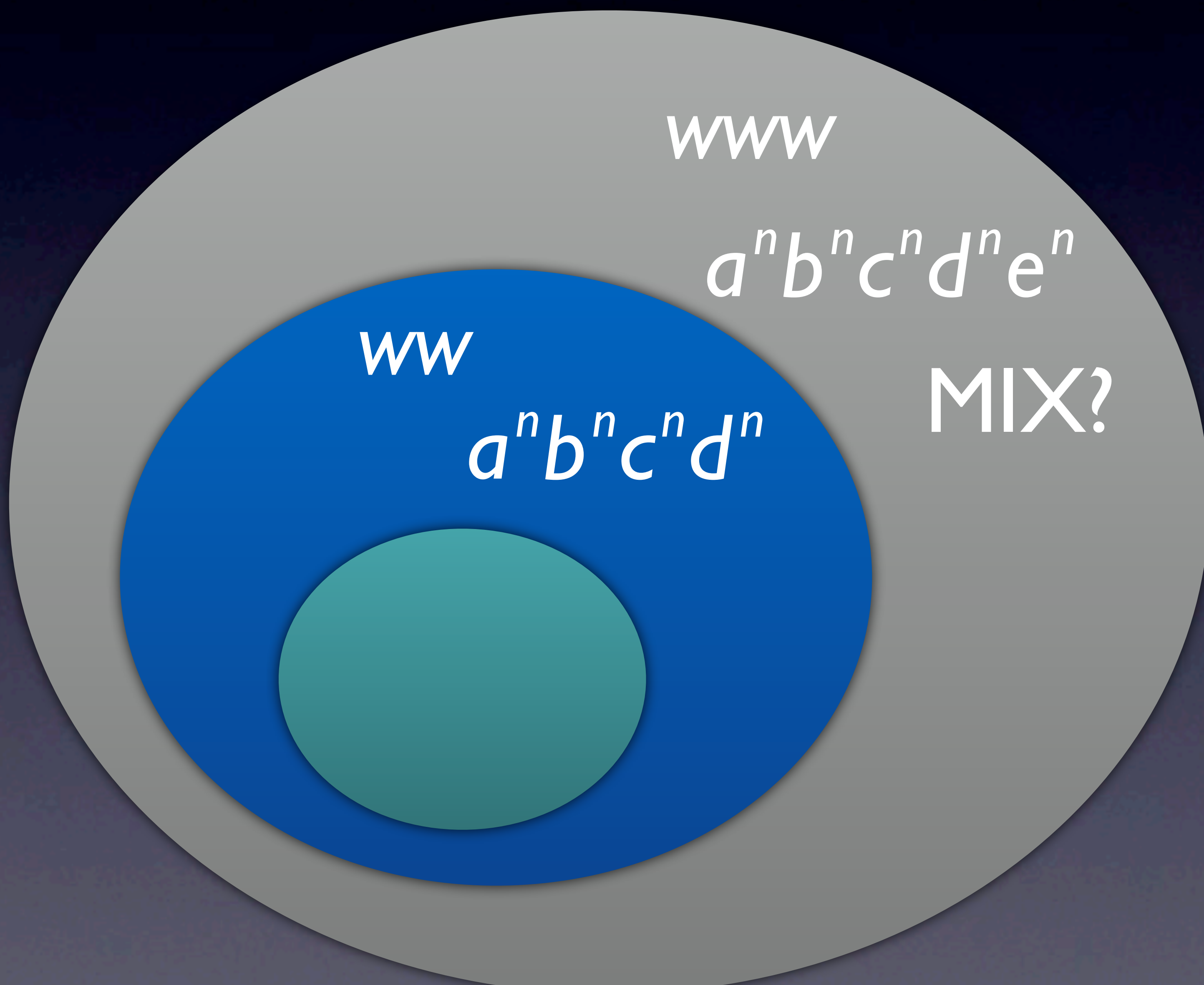
- Limited cross-serial dependencies
- constant growth
- polynomial parsing

"… roughly characterize a class of grammars (and associated languages) that are only slightly more powerful than context-free grammars (context-free languages)"

Joshi 1985

Called "mildly context-sensitive" because extends CFL only slightly
Three properties to be satisfied by grammars in between CF and CS

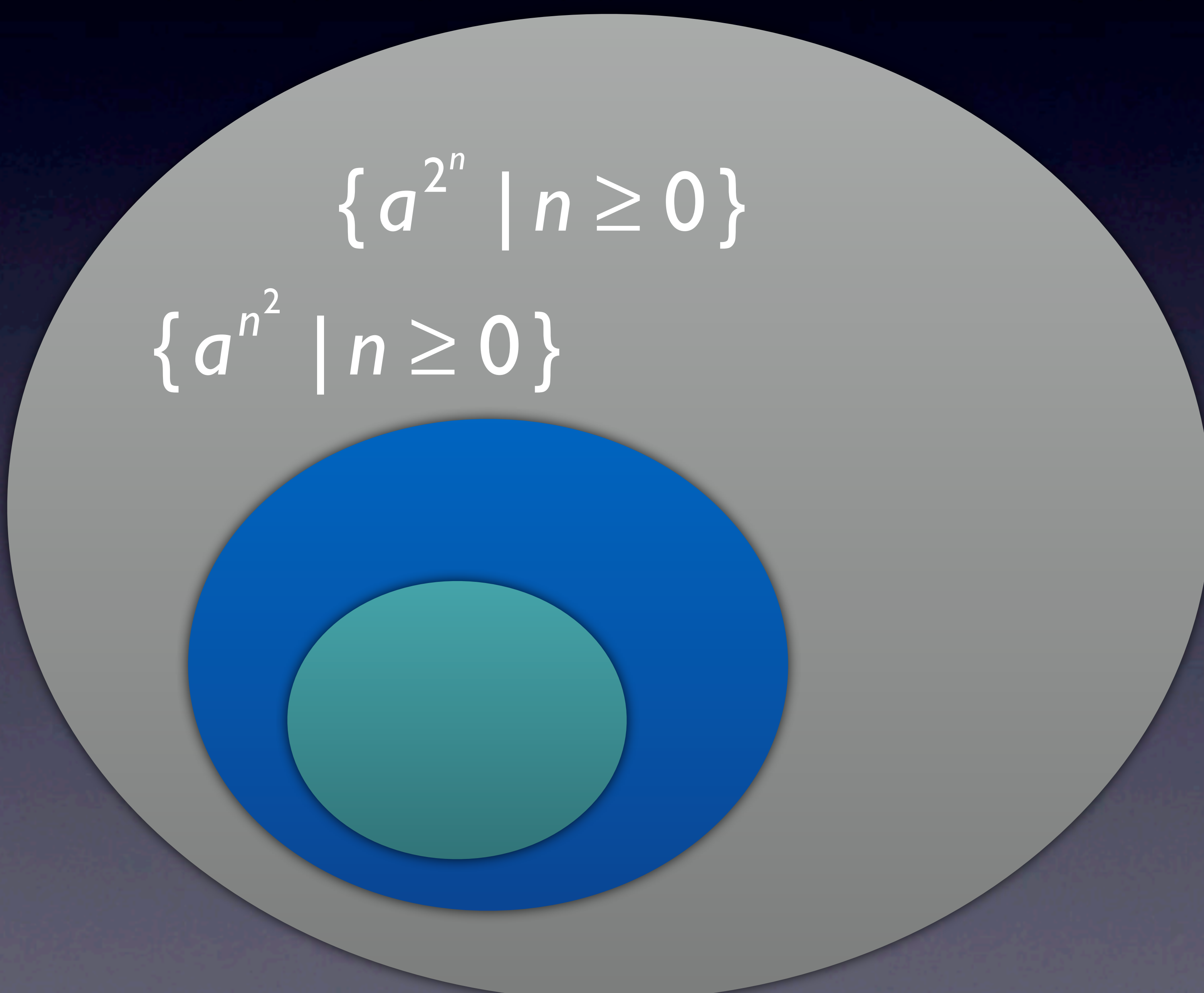
Limited cross-serial dependencies

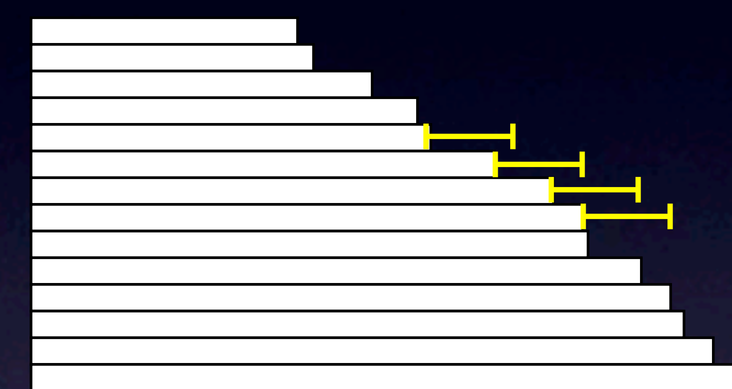


$$MIX = \{ w \in \{a, b, c\}^* \mid \#_a(w) = \#_b(w) = \#_c(w) \}$$

Informal, difficult to understand, meant to exclude these languages (at least in the case of the specific limit with TAGs)

Constant growth

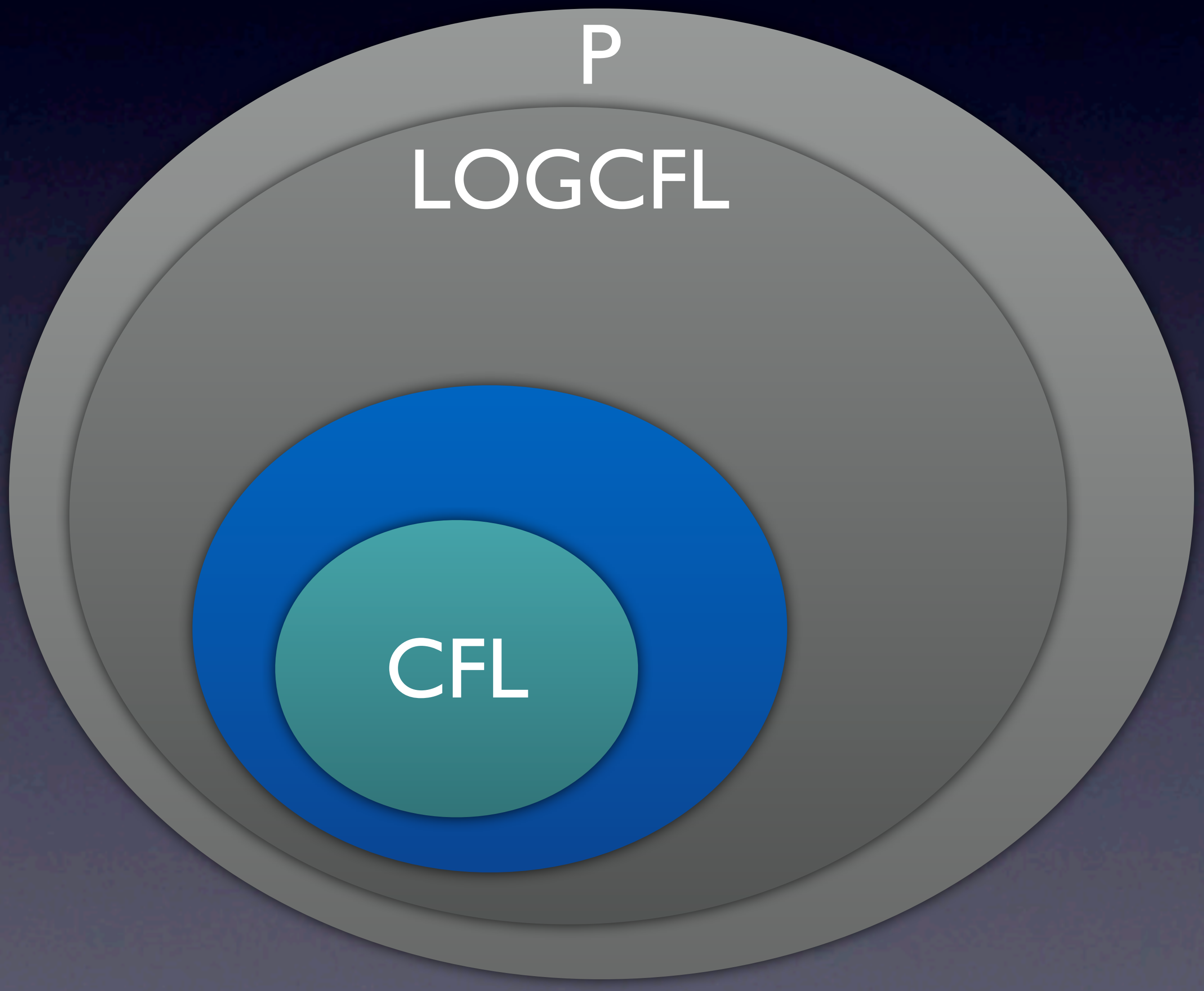




- Weakening of semilinearity

Simplification of semilinearity for expository purposes

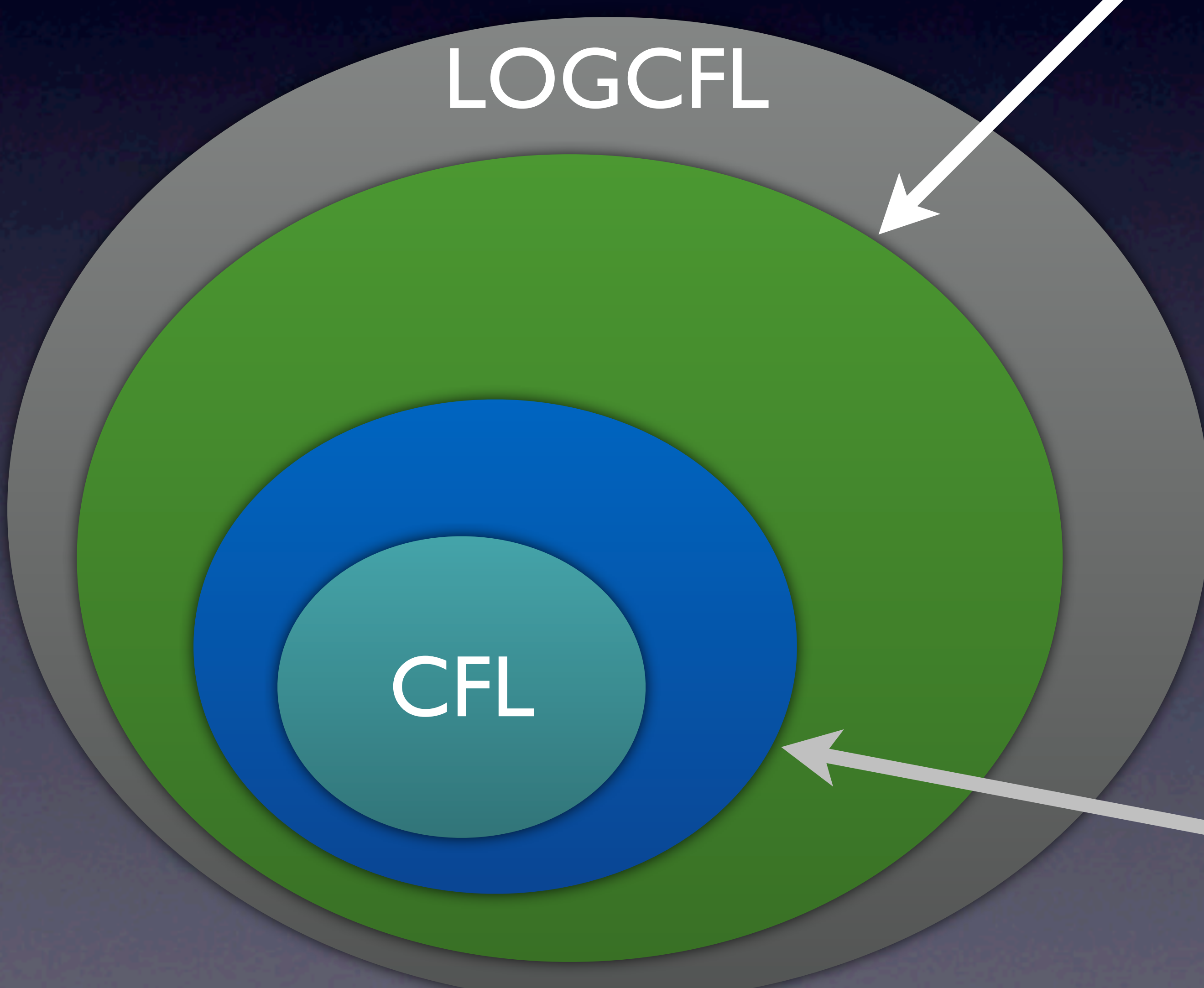
Polynomial parsing



- Containment in P
- Containment in LOGCFL

Much better to say LOGCFL than P

Another point of convergence



$MCTAG \equiv MCFG \equiv HR$

$\equiv OUT(DTWT)$

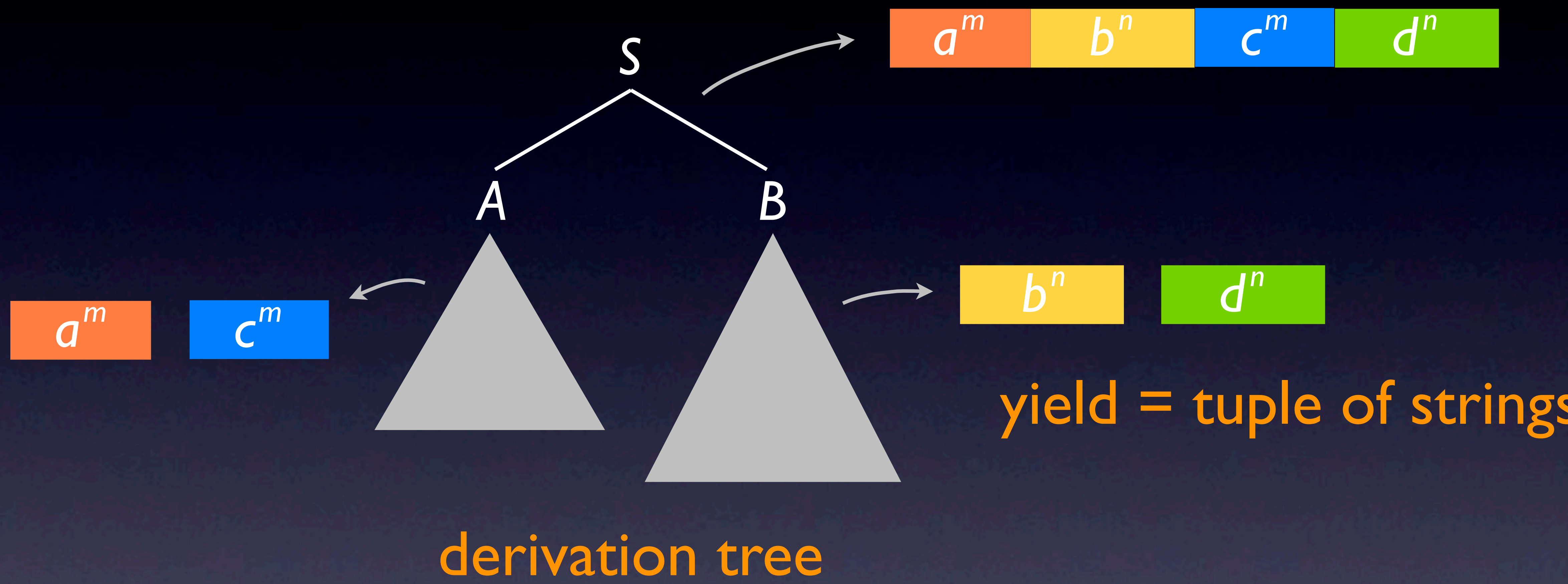
$\equiv yDT_{fc}(REGT) \equiv LUSCG$

$\equiv MG \equiv ACG_{(2,4)}$

$TAG \equiv CCG \equiv LIG \equiv HG$

The “convergence of mildly context-sensitive ...” orignally referred to TAG, CCG, LIG, HG, but in retrospect, ...
A greater number of equivalent formalisms, more diverse

Multiple context-free grammars



$$S(x_1 y_1 x_2 y_2) :\!- A(x_1, x_2), B(y_1, y_2).$$

$$A(\varepsilon, \varepsilon). \qquad B(\varepsilon, \varepsilon).$$

$$A(ax_1, cx_2) :\!- A(x_1, x_2). \qquad B(by_1, dy_2) :\!- B(y_1, y_2).$$

This is an example of a 2-MCFG.
An m-MCFG allows nonterminals to take up to m arguments.

m-multiple context-free grammars

Seki et al. 1991

$$N = \bigcup_{r \le m} N^{(r)}$$

$$G = (N, \Sigma, P, S)$$

ranked alphabet

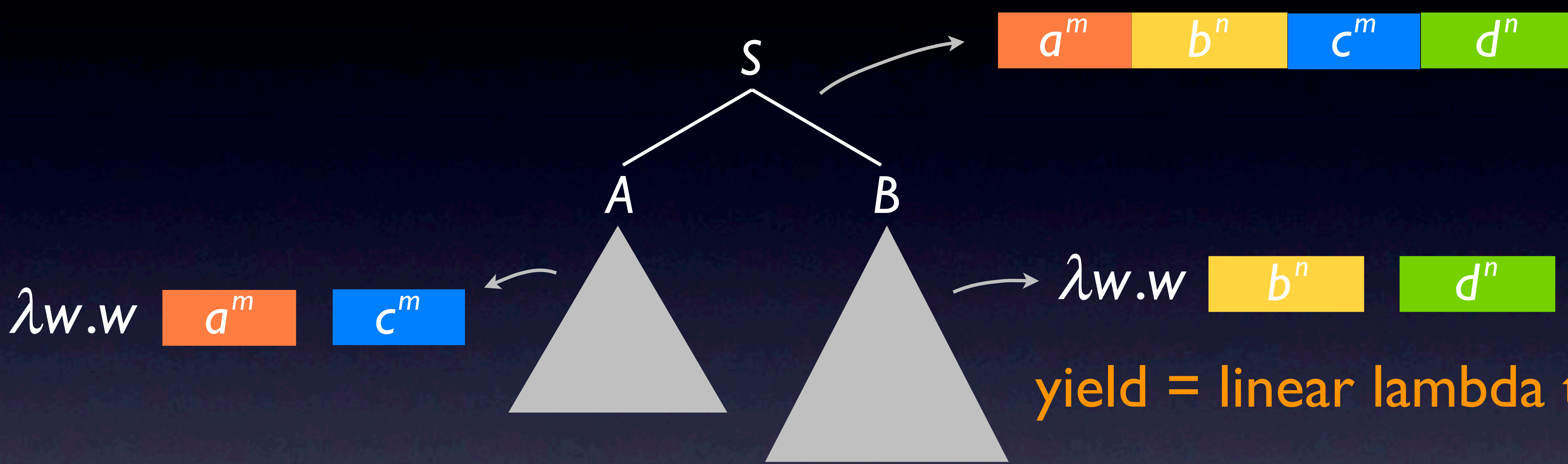
$$S \in N^{(1)}$$

$$B(t_1, \ldots, t_r) :\!- B_1(x_{1,1}, \ldots, x_{1,r_1}), \ldots, B_n(x_{n,1}, \ldots, x_{n,r_n}).$$

- $B \in N^{(r)}, B_i \in N^{(r_i)}$
- $t_1 \ldots t_r \in (\Sigma \cup X)^*$
- Each $x_{i,j}$ occurs at most once in $t_1 \ldots t_r$

$$L(G) = \{ w \in \Sigma^* \mid P \vdash S(w) \}$$

Second-order ACGs



yield = linear lambda term

$/abc/ = \lambda z.(a(b(cz)))$

derivation tree

$$S(\lambda z.X(\lambda x_1 x_2.Y(\lambda y_1 y_2.x_1(y_1(x_2(y_2 z)))))) :- A(X), B(Y).$$

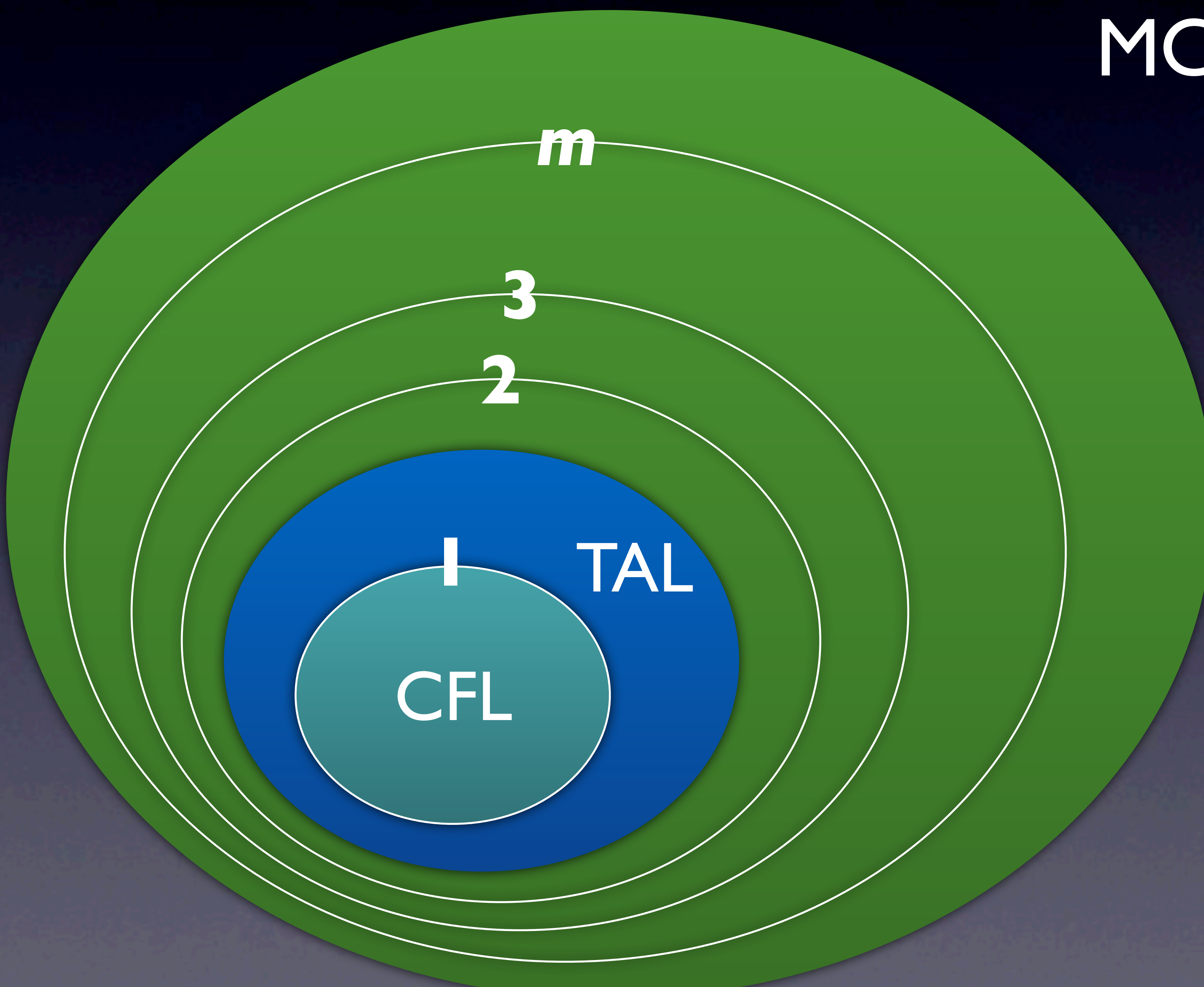
$$A(\lambda w.w(\lambda z.z)(\lambda z.z)).$$

$$A(\lambda w.X(\lambda x_1 x_2.w(\lambda z.a(x_1 z))(\lambda z.c(x_2 z)))) :- A(X).$$

This ACG encodes the example MCFG in the sense that there's a canonical correspondence between strings and linear lambda-terms
ACG_{(2,4)} because the type of the yield is up to fourth order

An infinite hierarchy

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



Each level of the hierarchy is equivalently defined by various other formalisms.
Not by ACG_{(2,4)}, however.

Consensus?

"MCTAGs also belong to the class of MCSGs and are in fact equivalent to LCFRSs."

Joshi, Vijay-Shanker, and Weir 1991

"The class of mildly context-sensitive languages seems to be most adequately approached by LCFRS."

Groenink 1997

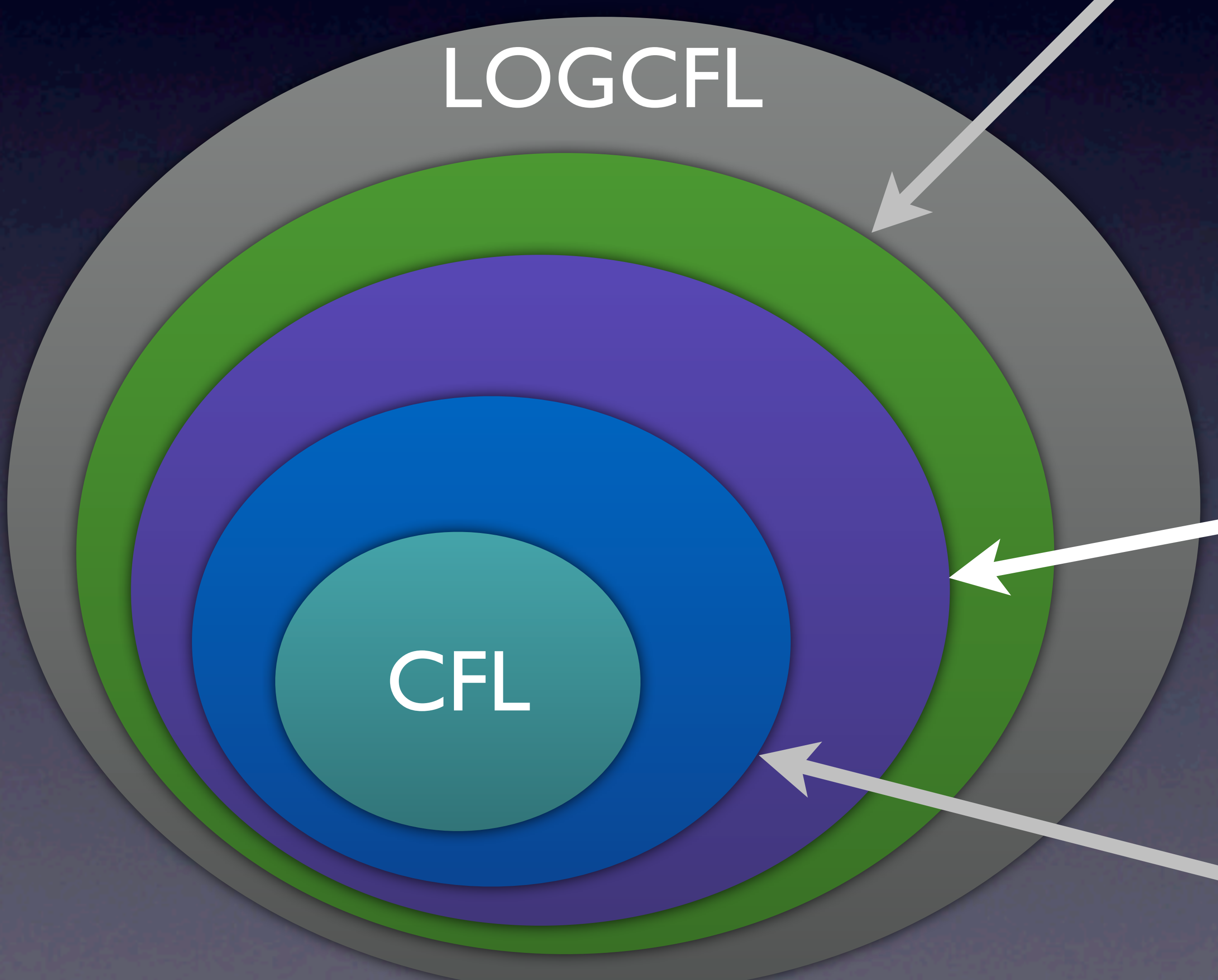
Because of the robustness of the class of MCFLs, a consensus seems to have emerged.
MCFL = MCSL

Consensus?

"Each MG … can be converted into a linear context-free rewriting system … In this sense MGs fall into the class of mildly context-sensitive grammars …"

Michaelis 1998

Yet another point of convergence



$\text{MCTAG} \equiv \text{MCFG} \equiv \text{HR} \equiv \text{OUT(DTWT)} \equiv \text{yDT}_{fc}\text{(REGT)} \equiv \text{LUSCG} \equiv \text{MG} \equiv \text{ACG}_{(2,4)}$

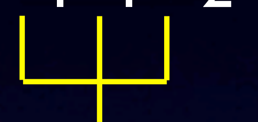
$\text{MCFG}_{wn} \equiv \text{CCFG} \equiv \text{Macro}_{nd} \equiv \text{ACG}_{(2,3)}$

$\text{TAG} \equiv \text{CCG} \equiv \text{LIG} \equiv \text{HG}$

The topic of this talk

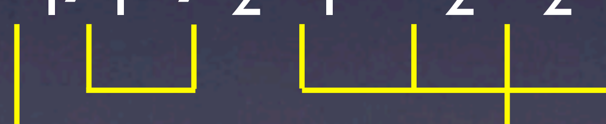
Well-nested MCFGs

✗ $S(x_1 y_1 x_2 y_2) :\!- A(x_1, x_2), B(y_1, y_2).$



✓ $S(x_1 y_1 y_2 x_2) :\!- A(x_1, x_2), B(y_1, y_2).$

✗ $C(x_1 y_1, y_2 z_1, z_2 x_2 z_3) :\!- A(x_1, x_2), B(y_1, y_2), C(z_1, z_2, z_3).$

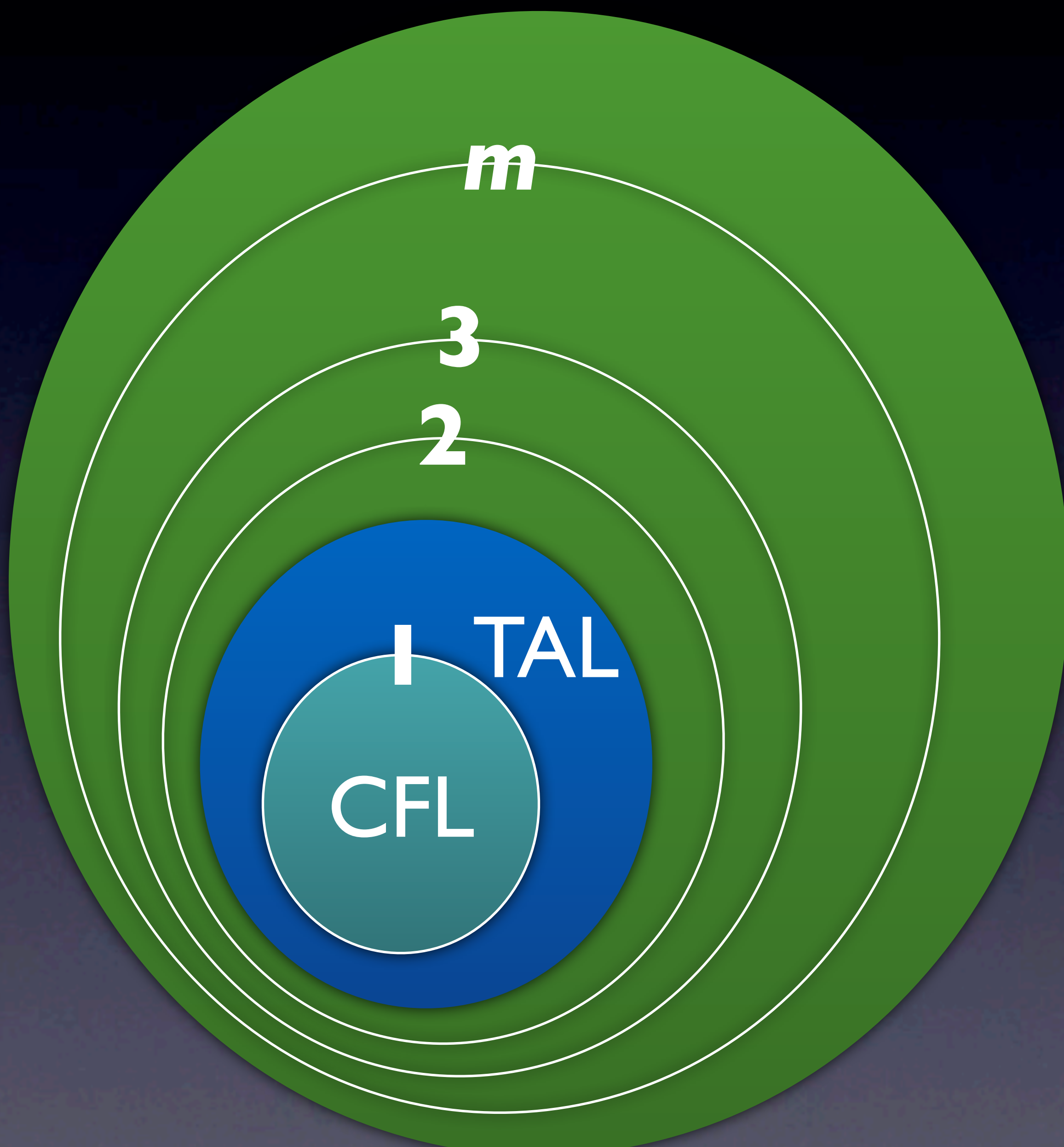


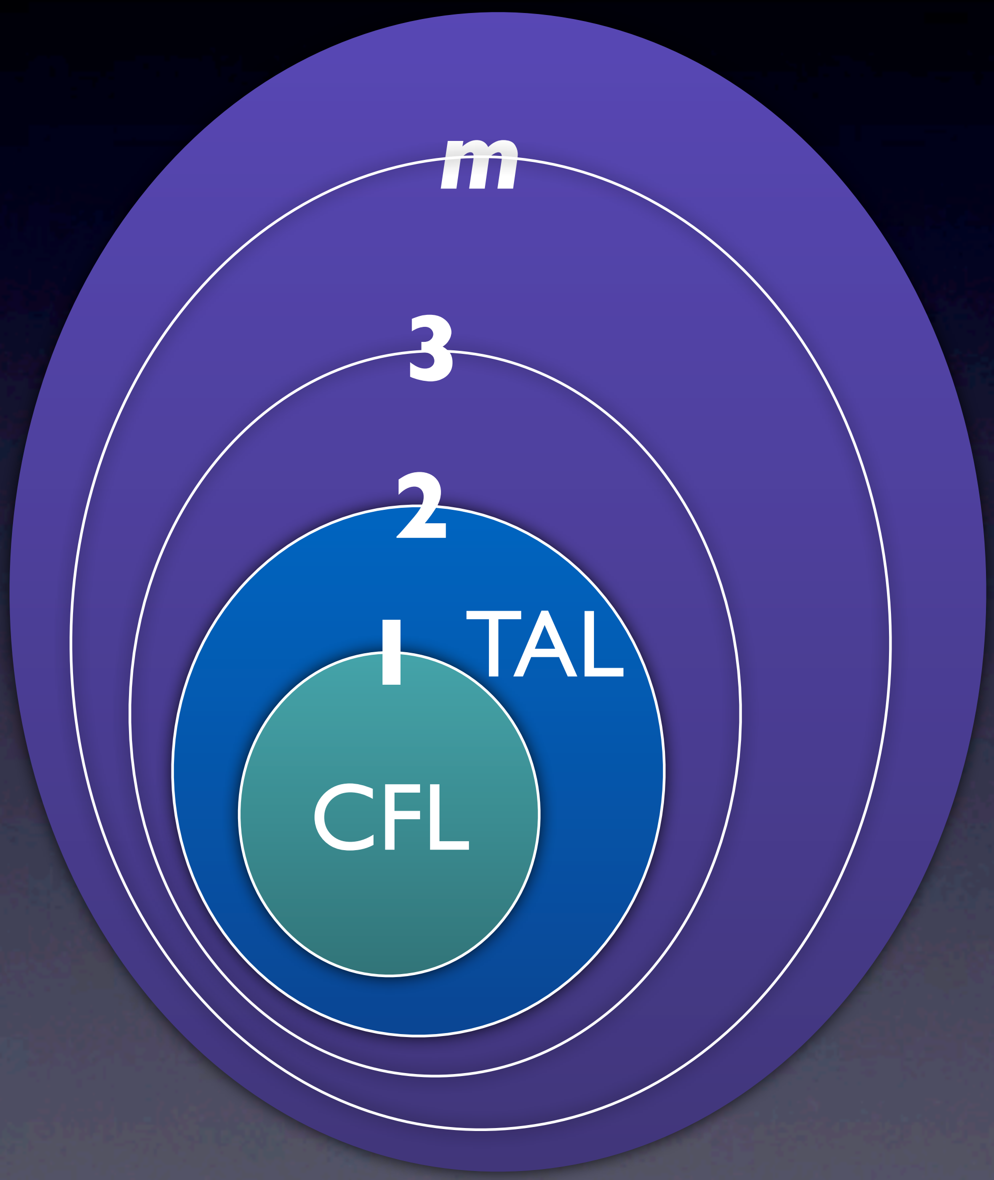
✓ $C(z_1 x_1, x_2 z_2, y_1 y_2 z_3) :\!- A(x_1, x_2), B(y_1, y_2), C(z_1, z_2, z_3).$

Cf. Kuhlmann 2007

Assume all rules are non-permuting.

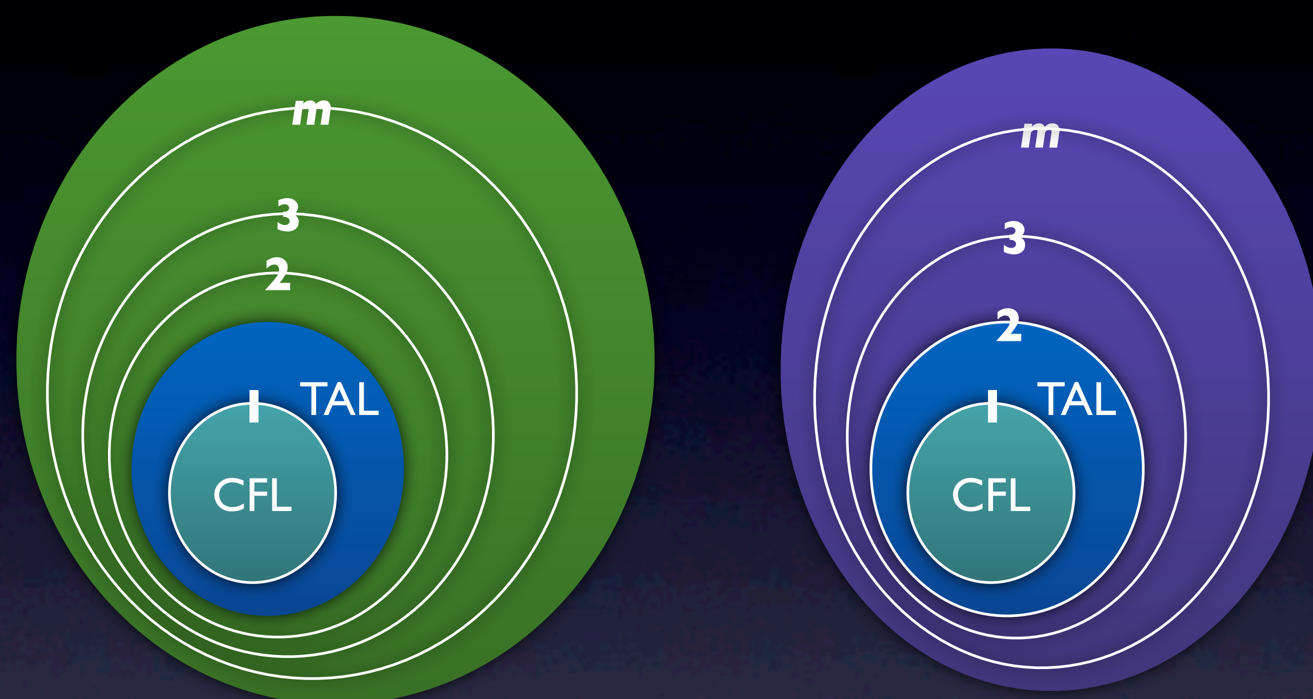
Two infinite hierarchies





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

Two infinite hierarchies



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

- How different are they?
- Which is a better formalization of “mildly context-sensitive grammar”?

m-MCFL vs. m-MCFL$_{\mathrm{wn}}$

$\mathrm{RESP}_2 = \{ a_1^i a_2^i b_1^j b_2^j a_3^i a_4^i b_3^j b_4^j \mid i, j \geq 0 \}$ Weir 1989

$\mathrm{RESP}_2 \in$ 2-MCFL – 2-MCFL$_{\mathrm{wn}}$ Seki et al. 1991

$\mathrm{RESP}_m = \{ a_1^i a_2^i b_1^j b_2^j \dots a_{2m-1}^i a_{2m}^i b_{2m-1}^j b_{2m}^j \mid i, j \geq 0 \}$

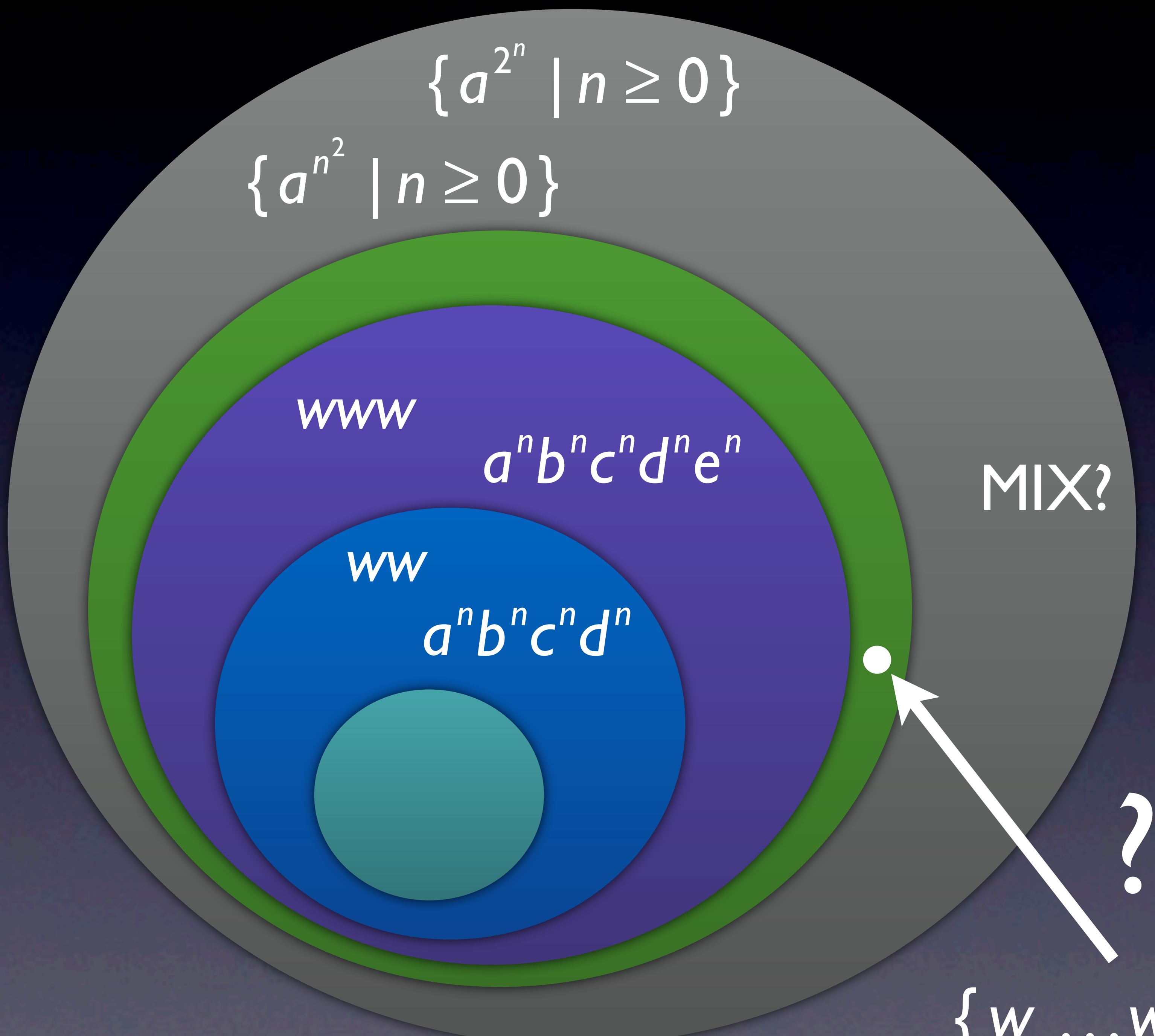
$\mathrm{RESP}_m \in m$-MCFL – m-MCFL$_{\mathrm{wn}}$ for $m \geq 2$

Seki and Kato 2008

$\mathrm{RESP}_m \in 2m$-MCFL$_{\mathrm{wn}}$

Separation is easy at each level

MCFL vs. MCFL_{wn}



Staudacher 1993
Michaelis 2005

$$\{w_1 \ldots w_n z_n w_n z_{n-1} \ldots z_1 w_1 z_0 w_1^R \ldots w_n^R \mid n \in \mathbb{N}, w_i \in \{a,b\}^+, z_n \ldots z_0 \in D_1^*\}$$

Much less clear with entire classes
D_1^* Dyck language

Limited cross-serial dependencies

6 Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions?

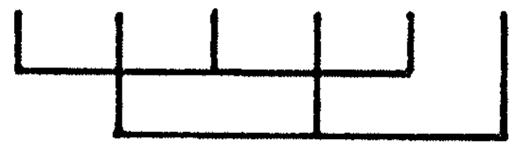
ARAVIND K. JOSHI

Joshi 1985

Which better captures mildly context-sensitivity

In a TAG we can characterize cross-serial dependencies between only two dependent sets and not more than two; hence, we cannot represent the cross-serial dependencies as in

(41) $a1\ a2\ b1\ b2\ c1\ c2$



(involving three dependent sets).

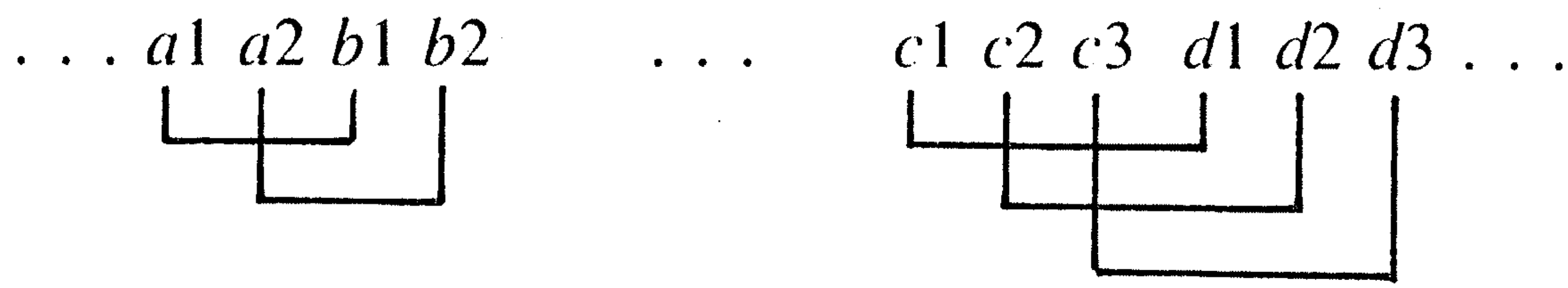
dependent sets → components of a derived tuple

$$S(x_1 x_2 x_3) :\!-\ A(x_1, x_2, x_3).$$
$$A(a x_1, b x_2, c x_3) :\!-\ A(x_1, x_2, x_3).$$
$$A(\varepsilon, \varepsilon, \varepsilon).$$

TAG is a 2-MCFG.

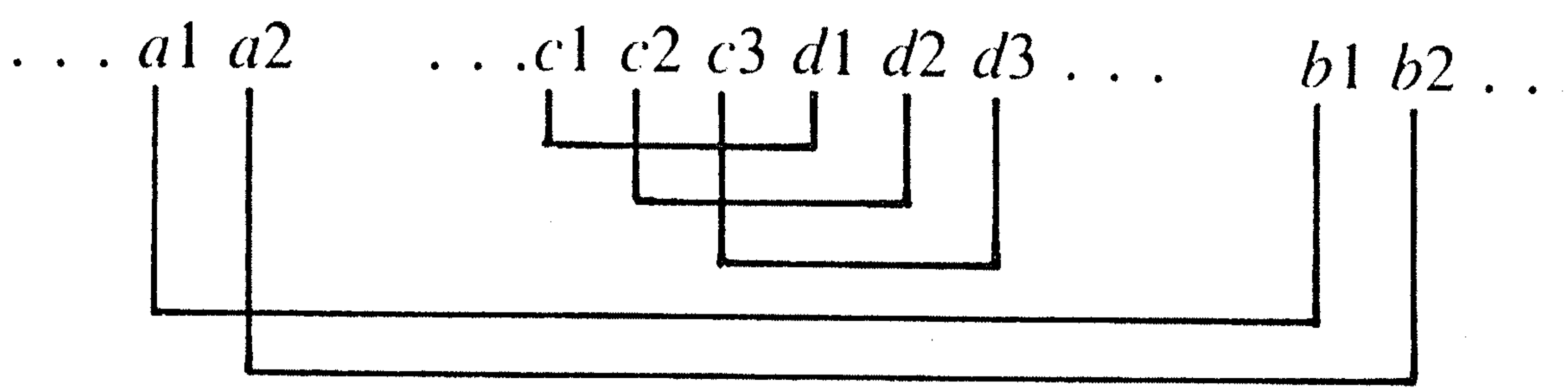
As long as the cross-serial dependencies involve only two dependent sets, as in the case of context-free grammars, we can have arbitrarily many such pairs of dependent sets, each with its cross-serial dependencies; however, any two pairs of such dependent sets are either disjoint or one is properly nested inside the other. Thus we have either

(42)



or

(43)



TAG is a well-nested MCFG.

Limited cross-serial dependencies

Thus TAGs allow a limited amount of cross-serial dependencies, and the dependent sets have the nesting properties as in the case of context-free grammars.

Limited amount of cross-serial dependencies
→ *m*-MCFG for some *m*

Nesting properties → well-nested MCFG

A mildly context-sensitive grammar
is a well-nested MCFG.

Pumpability

L is *k*-pumpable if there is a *p* satisfying the following condition:

for all $z \in L$ with $|z| \geq p$, *z* can be written as

$$z = u_0 v_1 u_1 \ldots u_{k-1} v_k u_k$$

in such a way that

$$0 < |v_1 \ldots v_k| \leq p, \text{ and}$$

$$u_0 v_1^i u_1 \ldots u_{k-1} v_k^i u_k \in L \text{ for all } i \geq 0.$$

Groenink 1997

Pumpability is related to semilinearity.

Pumping Lemma

Myth. If L is an m-MCFL, then L is $2m$-pumpable.

Radzinksi 1991, Groenink 1997, Kracht 2003

Theorem. If L is an m-$\mathrm{MCFL}_{\mathrm{wn}}$, then L is $2m$-pumpable.

Kanazawa 2009

Revised definition of mild context-sensitivity

- limited cross-serial dependencies, reduplication, and parallelism
- finite pumpability
- polynomial parsing

Groenink 1997

Question. Is every *m*-MCFL finitely pumpable?

Meant to capture a superclass of MCFL, but does it really?
Cross-serial + coordination

Polynomial parsing

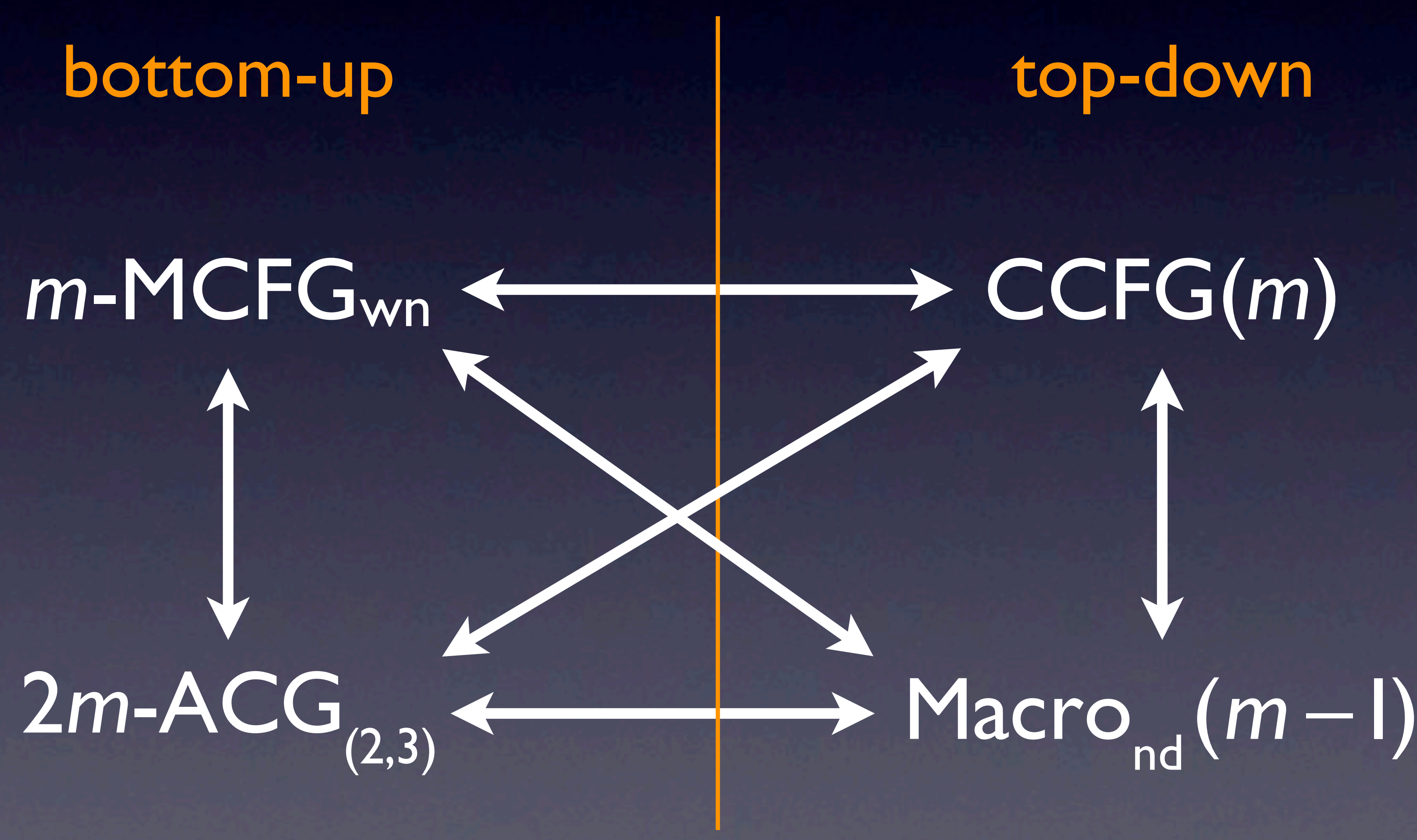
	fixed language recognition	universal recognition
CFG	LOGCFL-complete	P-complete
m-$MCFG_{wn}$	LOGCFL-complete	P-complete
m-MCFG	LOGCFL-complete	NP-complete ($m \geq 2$)
$MCFG_{wn}$	LOGCFL-complete	?
MCFG	LOGCFL-complete	PSACE-complete/ EXPTIME-complete

Mildly context-sensitive grammars

Is the notion of mild context-sensitivity most adequately captured by MCFGs?

Or by well-nested MCFGs?

The convergence of well-nested mildly context-sensitive grammar formalisms

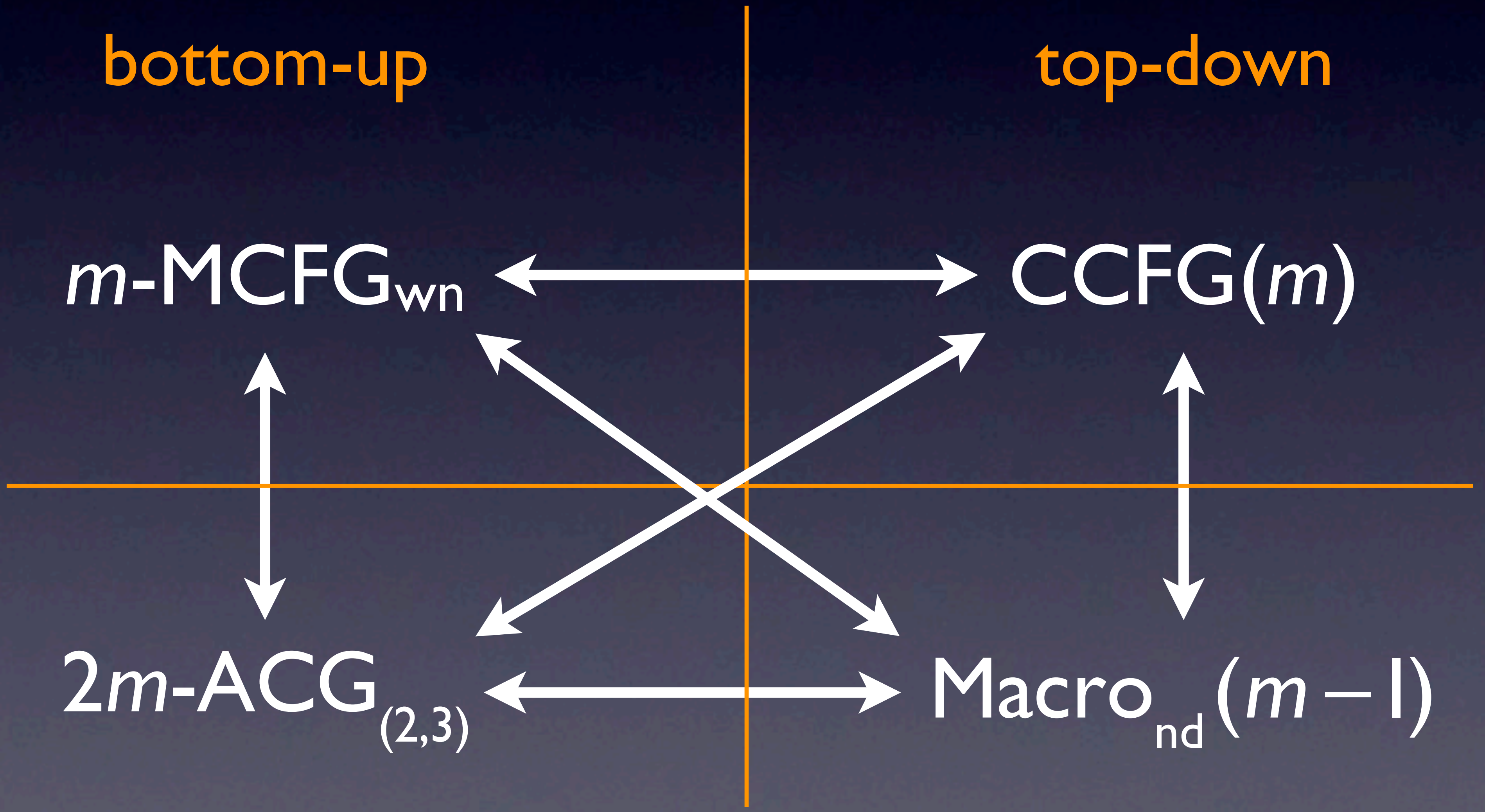


Top-down vs. bottom-up

$$A \rightarrow \boxed{B \quad B \quad C}$$

$$A\left(\boxed{X \quad Y \quad Z}\right) :- B(X), B(Y), C(Z).$$

Well-nested mildly context-sensitive grammar formalisms



Upper two non-permuting (viewed from the point of view of the lower two)

Well-nested mildly context-sensitive grammar formalisms

Rambow and Satta 1999

m-MCFG$_{wn}$ ⟷ CCFG(m)

m-MCFG ⟷ LUSCG(m)

$2m$-ACG$_{(2,3)}$

Macro$_{nd}(m-1)$

Well-nested mildly context-sensitive grammar formalisms

m-MCFG$_{wn}$

CCFG(m)

2m-ACG$_{(2,3)}$ ← Macro$_{nd}$($m-1$)

de Groote and Pogodalla 2004

Well-nested mildly context-sensitive grammar formalisms

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

$2m$-ACG$_{(2,3)}$ → Macro$_{nd}$($m-1$)

dG et al. 200x

Well-nested mildly context-sensitive grammar formalisms

m-MCFG$_{wn}$

CCFG(m)

2m-ACG$_{(2,3)}$

Macro$_{nd}$($m-1$)

Seki and Kato 2008

Well-nested mildly context-sensitive grammar formalisms

Do it this way, but no means the only easy way

m-MCFG$_{wn}$ → CCFG(m)

bottom-up ←

$$S(z_1 x_1 a x_2 z_2 b z_3) :- C(z_1, z_2, z_3), A(x_1, x_2).$$

$$C(z_1 x_1, x_2 z_2, y_1 y_2 z_3) :- A(x_1, x_2), B(y_1, y_2), C(z_1, z_2, z_3).$$

top-down →

$$S \to C^{[1]} A^{[1]} a A^{[2]} C^{[2]} b C^{[3]}$$

$$(C^{[1]}, C^{[2]}, C^{[3]}) \to (C^{[1]} A^{[1]}, A^{[2]} C^{[2]}, B^{[1]} B^{[2]} C^{[3]})$$

$$S \Rightarrow C^{[1]} A^{[1]} a A^{[2]} C^{[2]} b C^{[3]} \Rightarrow C^{[1]} A^{[1]} A^{[1]} a A^{[2]} A^{[2]} C^{[2]} b B^{[1]} B^{[2]} C^{[3]}$$

Have to keep track of links in the case of LUSCG, but not with CCFG

Extended Dyck set

$$N = \bigcup_r N^{(r)}$$

ranked alphabet

$ED(N, \Sigma)$ is the language defined by the following CFG:

$LL(1)$

$$S \rightarrow \varepsilon$$

$$S \rightarrow a\,S \quad (a \in \Sigma)$$

$$S \rightarrow S'\,S$$

$$S' \rightarrow B^{[1]}\,S\,B^{[2]} \ldots B^{[r-1]}\,S\,B^{[r]} \quad (B \in N^{(r)})$$

set of r parentheses

Coupled-context-free grammars

Hotz and Pitsch 1996

$$G = (N, \Sigma, P, S)$$

$$S \in N^{(1)}$$

$$(B^{[1]}, \ldots, B^{[r]}) \rightarrow (\beta_1, \ldots, \beta_r) \quad \beta_1 \ldots \beta_r \in ED(N, \Sigma)$$

- Sentential forms are in $ED(N, \Sigma)$
- $\alpha_0\, B_1\, \alpha_1 \, \ldots \, \alpha_{r-1}\, B_r\, \alpha_r \Rightarrow \alpha_0\, \beta_1\, \alpha_1 \, \ldots \, \alpha_{r-1}\, \beta_r\, \alpha_r$ if $\alpha_1, \ldots, \alpha_{r-1} \in ED(N, \Sigma)$.

CCFG(m) → Macro$_{\text{nd}}$(m−1)

top-down →

$$S \to C^{[1]}A^{[1]}aA^{[2]}C^{[2]}bC^{[3]}$$

$$(C^{[1]}, C^{[2]}, C^{[3]}) \to (C^{[1]}A^{[1]}, A^{[2]}C^{[2]}, B^{[1]}B^{[2]}C^{[3]})$$

$$S \Rightarrow C^{[1]}A^{[1]}aA^{[2]}C^{[2]}bC^{[3]} \Rightarrow C^{[1]}A^{[1]}A^{[1]}aA^{[2]}A^{[2]}C^{[2]}bB^{[1]}B^{[2]}C^{[3]}$$

top-down →

$$S \to C(A(a), b)$$

$$C(x_1, x_2) \to C(A(x_1), x_2B(\varepsilon))$$

$$S \Rightarrow C(A(a), b) \Rightarrow C(A(A(a)), bB(\varepsilon))$$

non-duplicating, non-deleting, non-permuting

Nested terms

$NT(N, \Sigma)$ is the language defined by the left CFG below:

$$S \to \varepsilon$$
$$S \to a\,S \quad (a \in \Sigma)$$
$$S \to S'\,S$$
$$S' \to B(\underbrace{S,\ldots,S}_{r \text{ times}}) \quad (B \in N^{(r)})$$

$NT(N, \Sigma)$

$LL(1)$

$$S \to \varepsilon$$
$$S \to a\,S \quad (a \in \Sigma)$$
$$S \to S'\,S$$
$$S' \to B^{[1]}\,S\,B^{[2]}\ldots B^{[r]}S\,B^{[r+1]}$$

$ED(\mathrm{inc}(N), \Sigma)$

NT(N,Sigma) and ED(inc(N),Sigma) isomorphic

Non-duplicating macro grammars

Fischer 1968

$$G = (N, \Sigma, P, S)$$

$$S \in N^{(0)}$$

$$B(x_1, \ldots, x_r) \to \beta$$

- $\beta \in NT(N, \Sigma \cup \{x_1, \ldots, x_r\})$
- Non-duplicating if each x_i occurs in β at most once
- Sentential forms are in $NT(N, \Sigma)$
- $\alpha_0\, B(\alpha_1, \ldots, \alpha_r)\ \alpha_{r+1} \Rightarrow \alpha_0\, \beta[x_i := \alpha_i]\ \alpha_{r+1}$

$\text{Macro}_{\text{nd}}(m-1) \rightarrow$ 2m-ACG$_{(2,3)}$

de Groote and Pogodalla 1994

top-down

$$S \rightarrow C(A(a), b)$$

$$C(x_1, x_2) \rightarrow C(A(x_1), x_2 B(\varepsilon))$$

$$S \Rightarrow C(A(a), b) \Rightarrow C(A(A(a)), bB(\varepsilon))$$

bottom-up

$$S(Z(Ya))b) \coloneq C(Z), A(X).$$

$$C(\lambda x_1 x_2 . Z(X x_1)(\lambda z. x_2 (Y(\lambda z.z)))) \coloneq C(Z), A(X), B(Y).$$

Linear lambda-terms over Σ

$$\vdash a : o \to o \quad (a \in \Sigma)$$

$$x : \alpha \vdash x : \alpha$$

$$\frac{\Gamma \vdash M : \alpha \to \beta \quad \Delta \vdash N : \alpha}{\Gamma, \Delta \vdash MN : \beta} \ (\mathrm{dom}(\Gamma) \cap \mathrm{dom}(\Delta) = \emptyset)$$

$$\frac{\Gamma, x : \alpha \vdash M : \beta}{\Gamma \vdash \lambda x.M : \alpha \to \beta}$$

Second-order ACGs in (2,3) of width *m*

$$G = (N, \Sigma, \tau, P, S)$$

$$\tau : N \to Types$$

$$\tau(S) = o \to o$$

$$\mathrm{ord}(\tau(B)) \leq 3 \quad \text{for all } B \in N$$

$$|\tau(B)| \leq m \quad \text{for all } B \in N$$

$$B(M) :- B_1(X_1), \ldots, B_n(X_n)$$

where $X_1 : \tau(B_1), \ldots, X_n : \tau(B_n) \vdash M : \tau(B)$

Normal form for 2m-ACG$_{(2,3)}$

$$\tau(B) = (o \to o)^r \to (o \to o) \quad (0 \le r \le m-1)$$

$P \vdash B(M)$ implies

$$M =_\beta \lambda y_1 \ldots y_r z. /w_1/\, (y_1(\ldots(y_r(/w_{r+1}/\, z))\ldots))$$

for some $w_1, \ldots, w_{r+1} \in \Sigma^*$

Easy to go to Macro_{nd}(m-1) from here

$2m$-ACG$_{(2,3)}$ → m-MCFG$_{wn}$

Cf. Kanazawa and Salvati 2007

$$B(M) :- B_1(X_1), \ldots, B_n(X_n)$$

$$\tau(B) = (o \to o)^r \to (o \to o)$$

$$\tau(B_i) = (o \to o)^{r_i} \to (o \to o)$$

$$B(t_1, \ldots, t_{r+1}) :- B_1(x_{1,1}, \ldots, x_{1,r_1+1}), \ldots, B_n(x_{n,1}, \ldots, x_{n,r_n+1})$$

where $M[X_i := \lambda y_{i,1} \ldots y_{i,r_i} z.x_{i,1}(y_{i,1}(\ldots(y_{i,r_i}(x_{i,r_i+1} z))\ldots))]$

$$=_\beta \lambda y_1 \ldots y_r z. /t_1/ (y_1(\ldots(y_r(/t_{r+1}/ z))\ldots))$$

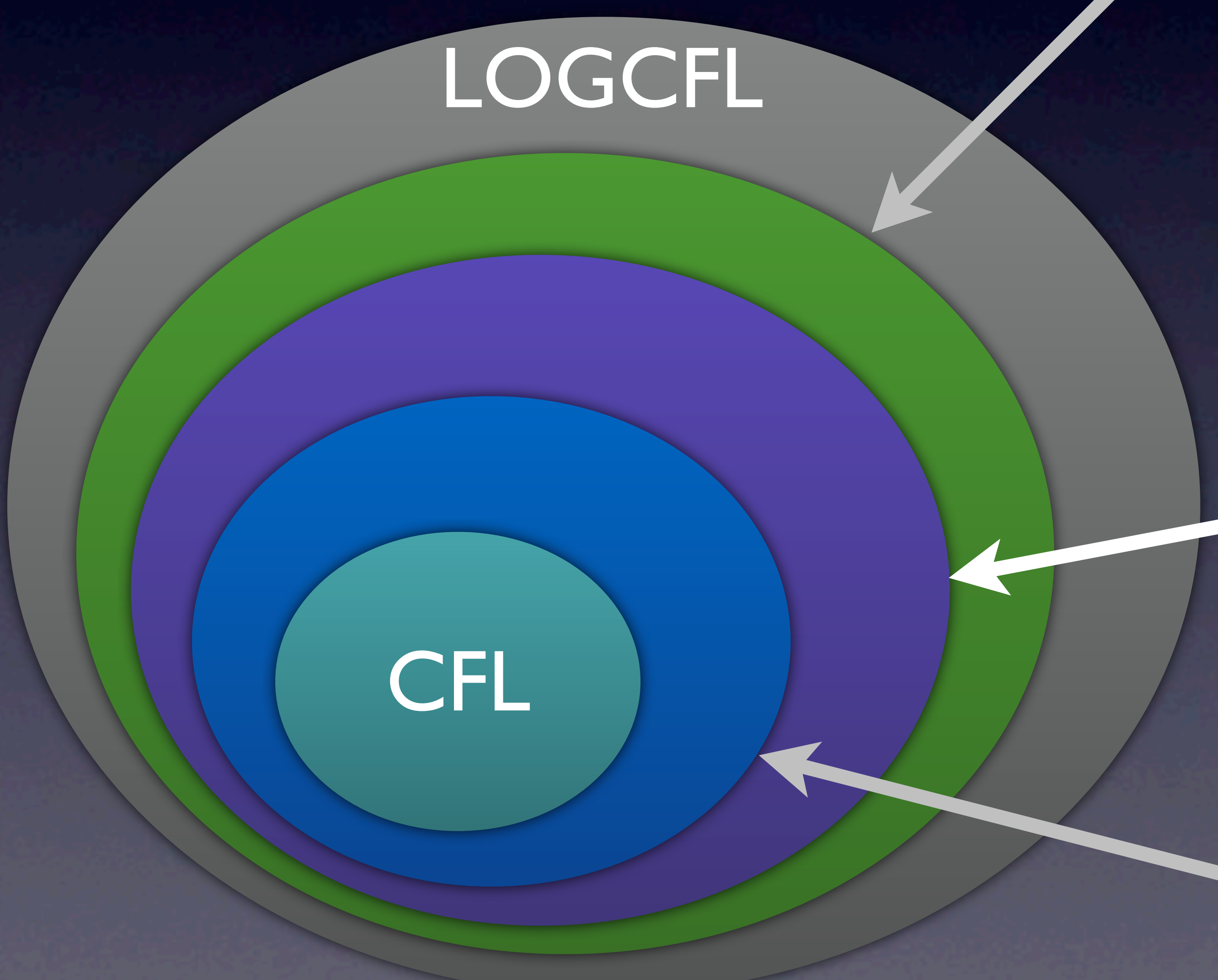
Similar transformation to the general case
The result well-nested

Well-nested mildly context-sensitive grammar formalisms

All similar to each other, “context-free”

2m-ACG_{(2,3)} not defined to be similar, but found to be so by analysis

Yet another point of convergence



MCTAG ≡ MCFG ≡ HR

≡ OUT(DTWT)

≡ yDT_{fc}(REGT) ≡ LUSCG

≡ MG ≡ $ACG_{(2,4)}$

$MCFG_{wn}$ ≡ CCFG ≡ $Macro_{nd}$

≡ $ACG_{(2,3)}$

TAG ≡ CCG ≡ LIG ≡ HG

Not as robust as MCFL, but MCFL_{wn} also interesting