Relating dominance formalisms

Alexander Koller and Owen Rambow
Columbia University

Formal Grammars
Dublin
5 August 2007

Outline

- Two dominance-based formalisms
  - dominance graphs
    (built for semantics)
  - vector grammars with dominance links
    (built for syntax)
- Encoding of dominance graphs into grammars
  - configurations = parse trees
  - charts are isomorphic

Normal dominance graphs

(Egg et al. 2001; Althaus et al. 2003; ...)

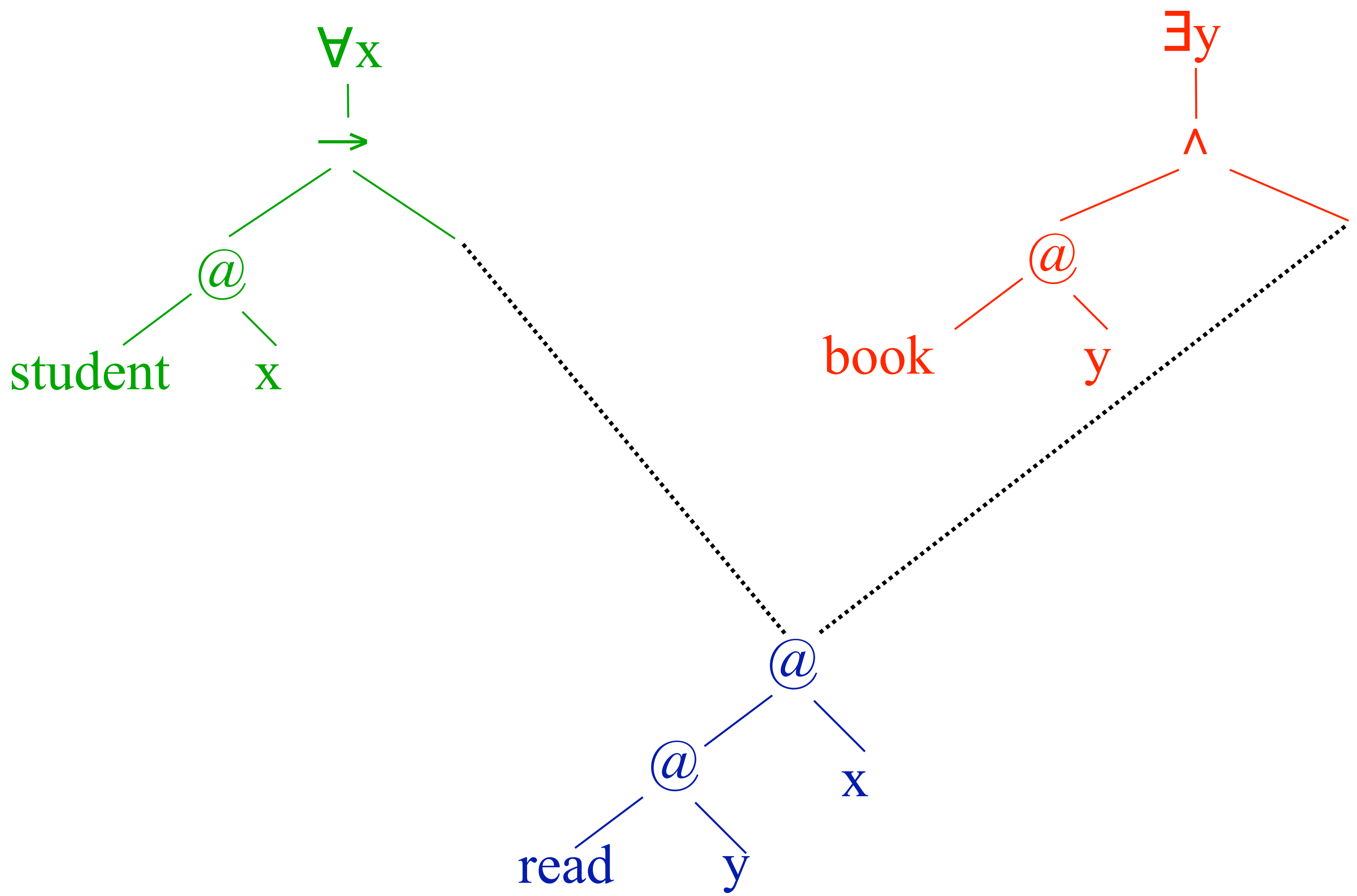
- Developed for scope underspecification.
- Read semantic representations (e.g. formulas of predicate logic, DRSs, etc.) as trees.
- Describe these trees using graphs that can be embedded into them.
- Equivalent to normal dominance constraints, i.e. both graph view and logic view available.
- Dominance constraints are a fragment of CLLS (the Constraint Language for Lambda Structures).

Describe Trees Using Graphs

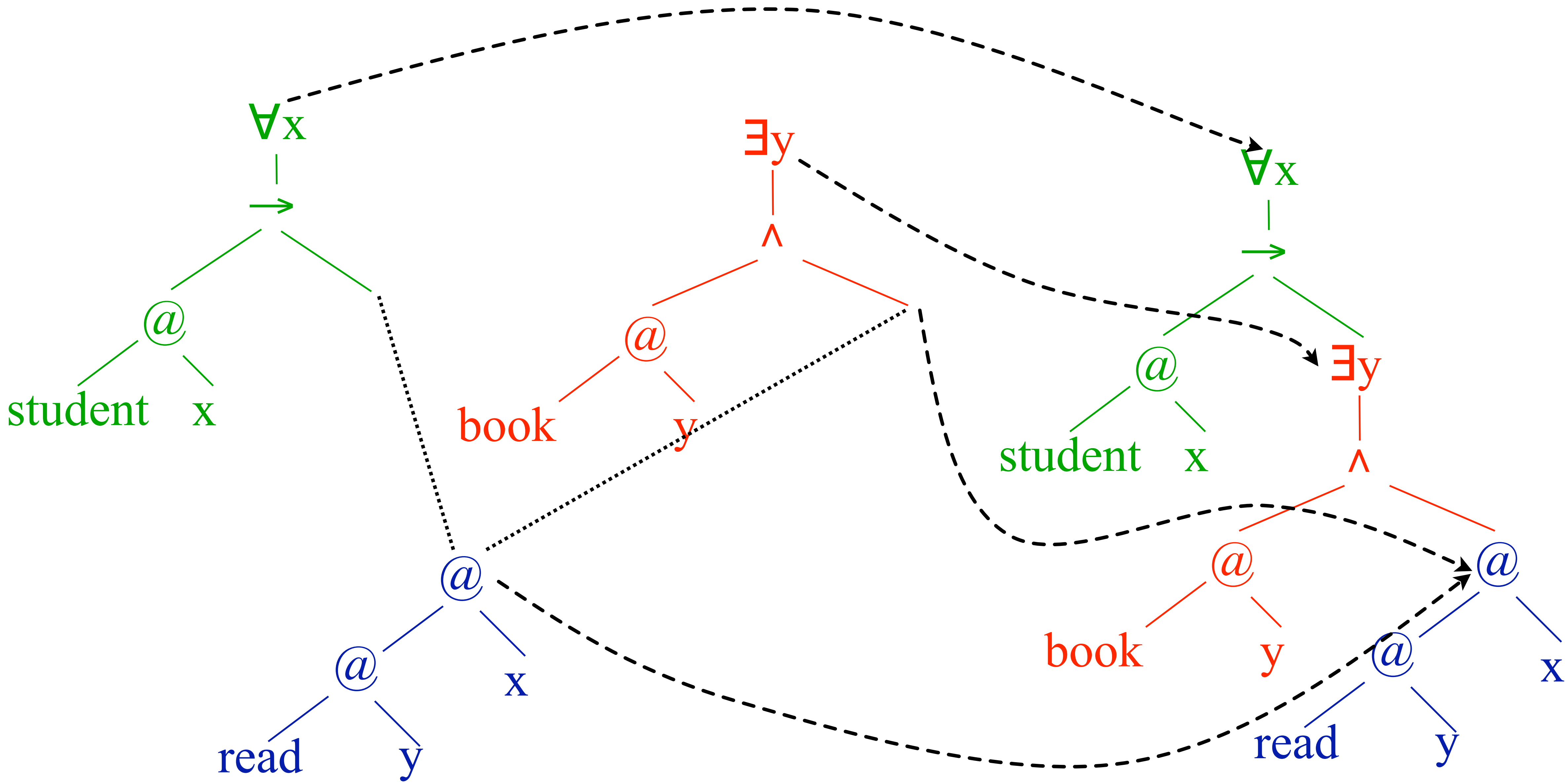
"Every student reads a book."

$\exists y\ book(y) \wedge \forall x\ student(x) \rightarrow read(y)(x)$

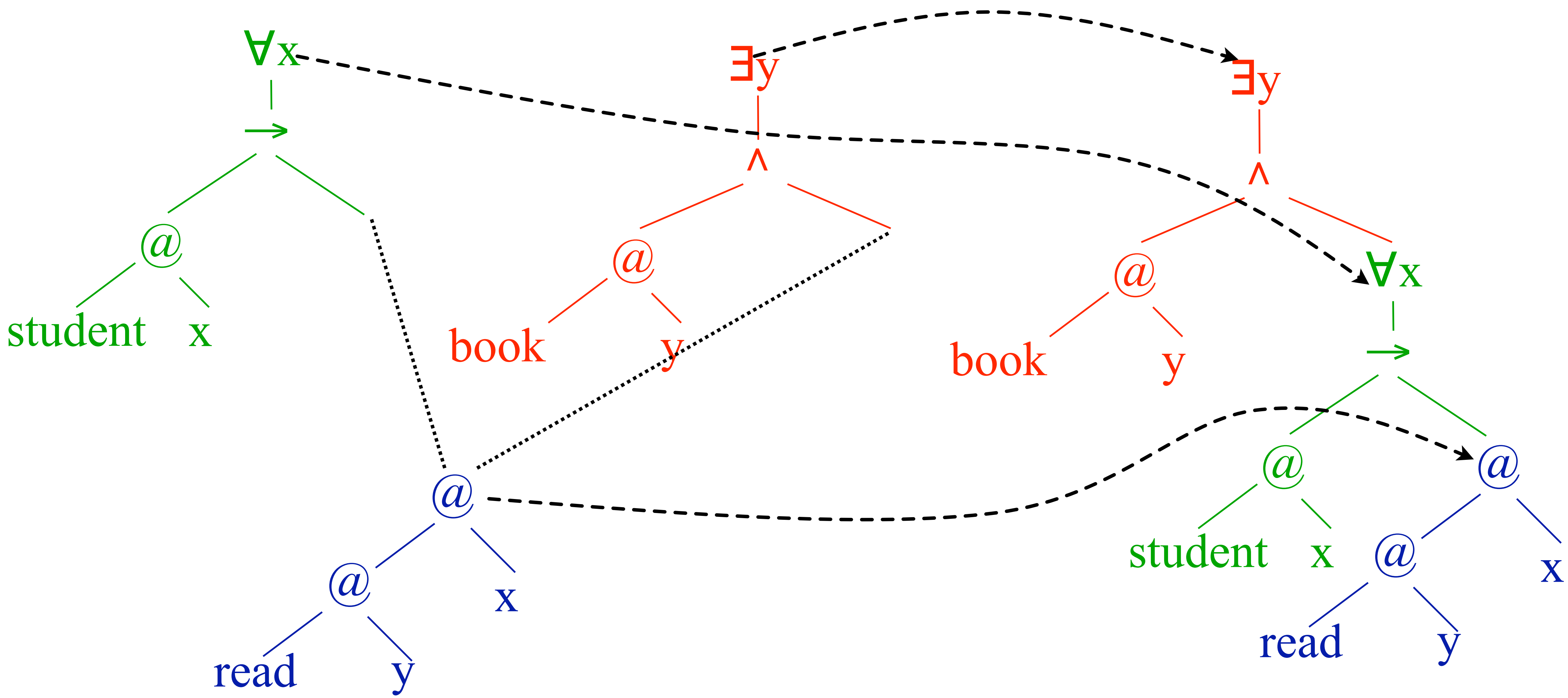
$\forall x\ student(x) \rightarrow \exists y\ book(y) \wedge read(y)(x)$



Describe Trees Using Graphs

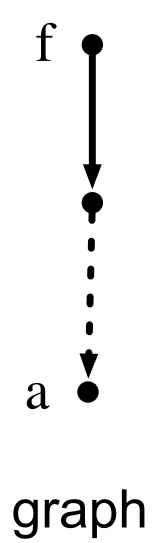


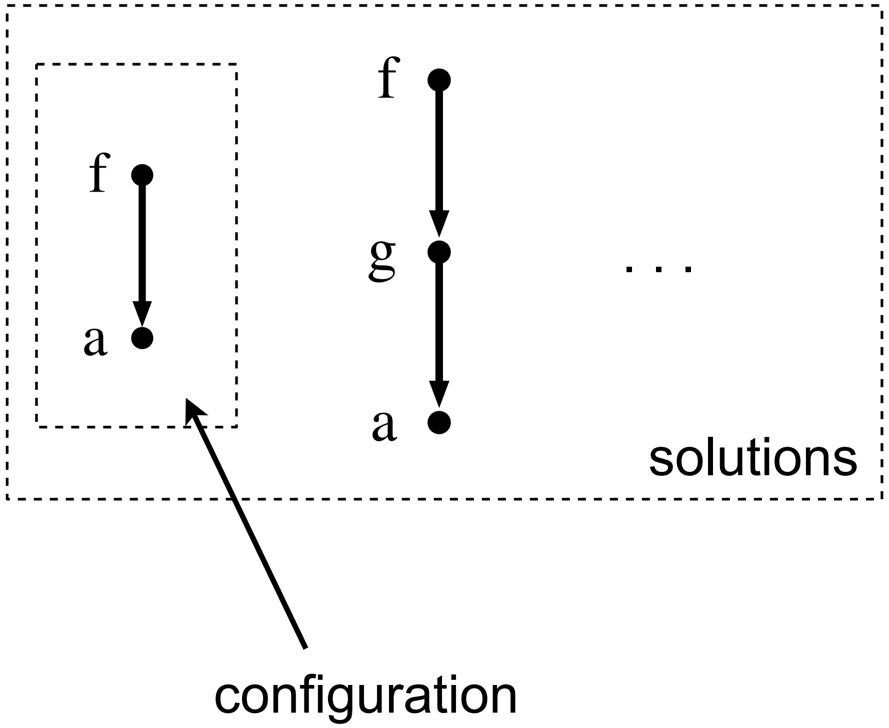
Describe Trees Using Graphs



Solutions vs. configurations

- Dominance graphs have
  - solutions: any tree into which graph can be embedded, respecting dominance edges
  - configurations: solutions whose nodes are all images of labelled nodes in the graph





Solutions vs. configurations

- Some graphs have solutions but no configurations:

- Solvability can be decided in linear time; configurability is NP-complete.
- For hypernormally connected dominance graph, solvability and configurability are equivalent.

UVG-DL grammars

(Rambow 1994; ...)

- Grammar G = (N,T,V,S) where V is a set of vectors, and each vector is a set of context-free production rules.
- Within each vector, may specify dominance links between nonterminal occurrences.
- Derivation: Like context-free derivation, but
  - the multiset of production rule applications must be the multiset union of some vectors
  - dominance links are respected in the parse tree

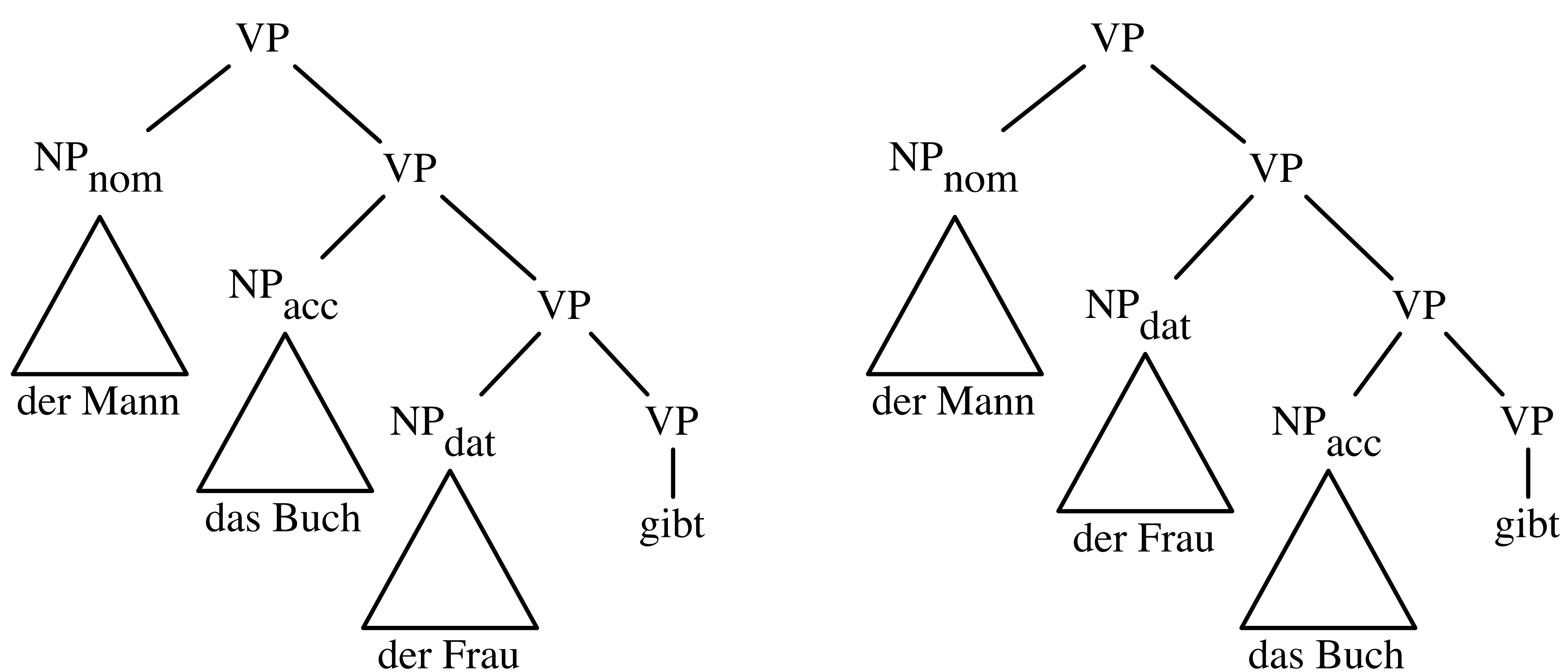
UVG-DL: An example

grammar:

$\{NP_{nom} \rightarrow \text{der Mann}\}$ $\{NP_{acc} \rightarrow \text{das Buch}\}$ $\{NP_{dat} \rightarrow \text{der Frau}\}$

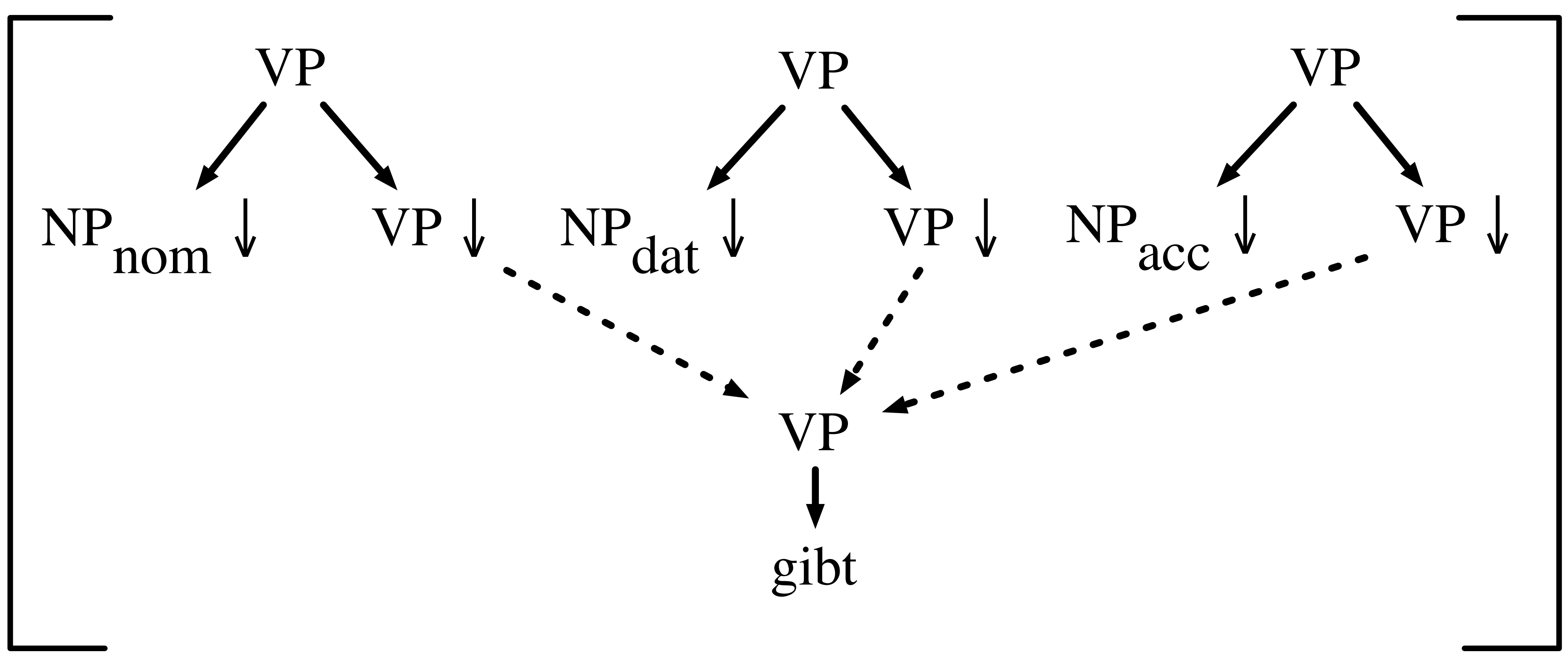
$\{ VP \rightarrow NP_{nom}\ VP,\ VP \rightarrow NP_{dat}\ VP,\ VP \rightarrow NP_{acc}\ VP,\ VP \rightarrow \text{gibt} \}$

two (of six) parse trees:



Think of it in terms of TAG

$\{ VP \rightarrow NP_{nom}\ VP,\ VP \rightarrow NP_{dat}\ VP,\ VP \rightarrow NP_{acc}\ VP,\ VP \rightarrow gibt \}$



Relationship between these formalisms?

semantics tradition

Bos 96

Egg et al. 98

Althaus et al. 01

Koller et al. 03

Fuchss et al. 04

Flickinger et al. 05

Koller & Thater 05, 06

...

?

syntax tradition

Vijay-Shanker 92

Rambow 94

Becker & Rambow 95

Rambow & Satta 96

Rambow et al. 01

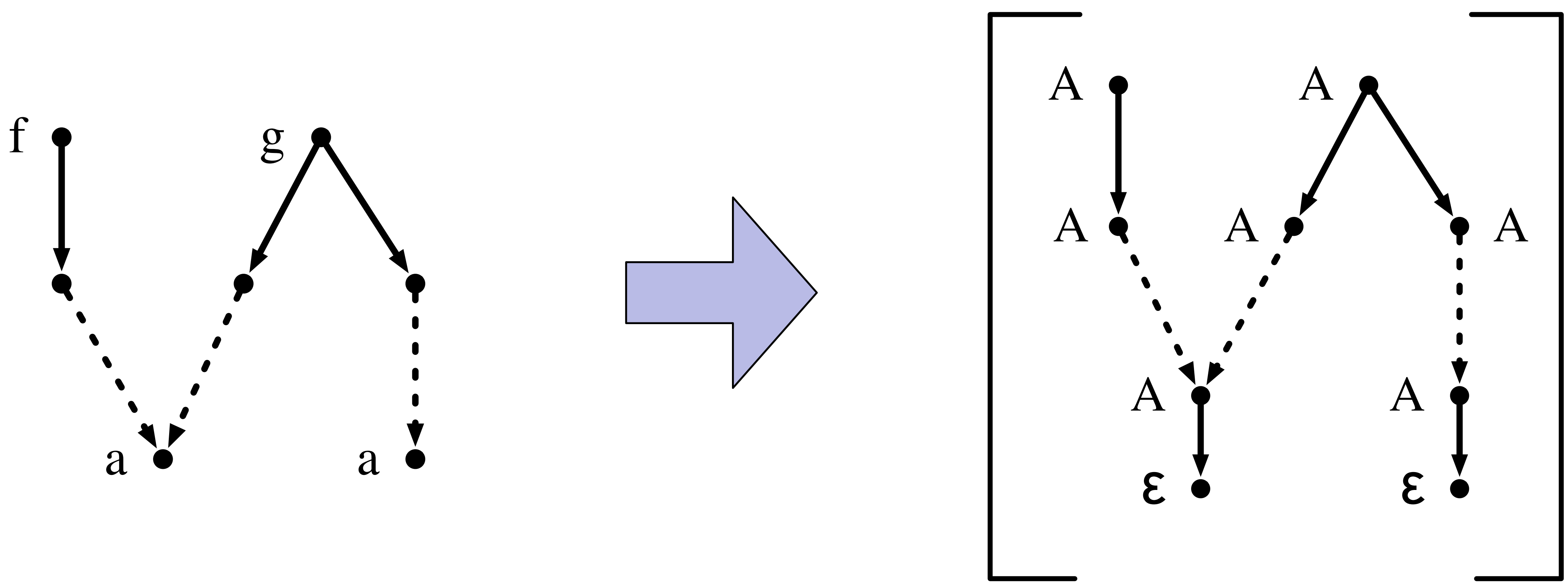
Rogers 03

Gerdes & Kahane 06

...

Dominance graphs as UVG-DL grammars

- Result 1: Can encode every normal dominance graph as an UVG-DL grammar such that configurations = parse trees of ε.

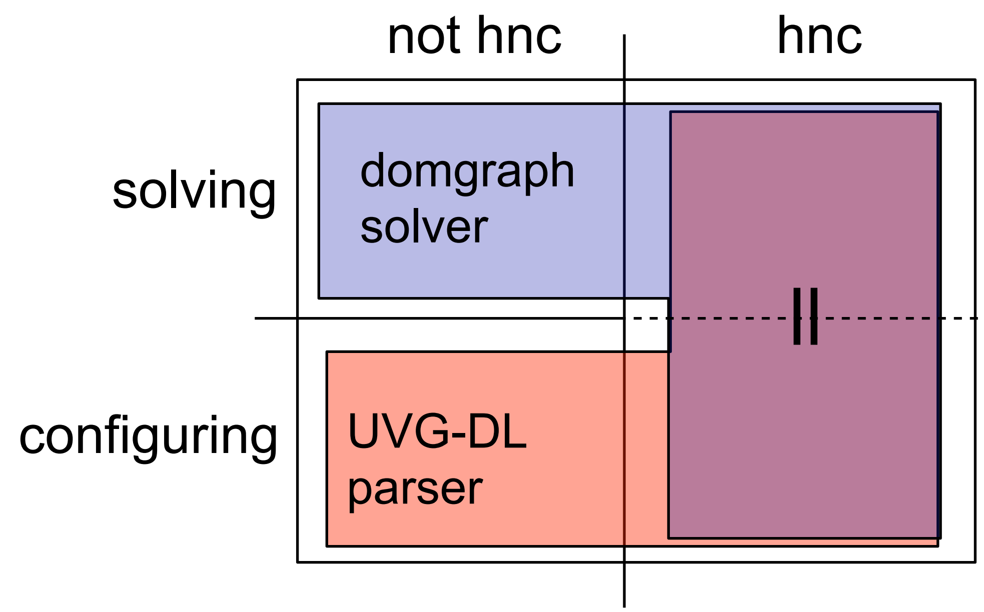


Harvesting

- Some consequences:
  - Word problem of UVG-DL is NP-complete if the grammar is part of the input. (Previously: best algorithms polynomial in input size, exponential in grammar size)
  - UVG-DL parser can be used to compute configurations of dominance graph. (Previously: No explicit algorithms for computing configurations.)

Relating the charts

- For hypernormally connected graphs, we now have two independent algorithms for computing configurations of dominance graphs:
  - UVG-DL parser (see above)
  - standard solver for dominance graphs



- Let's compare the way they work!

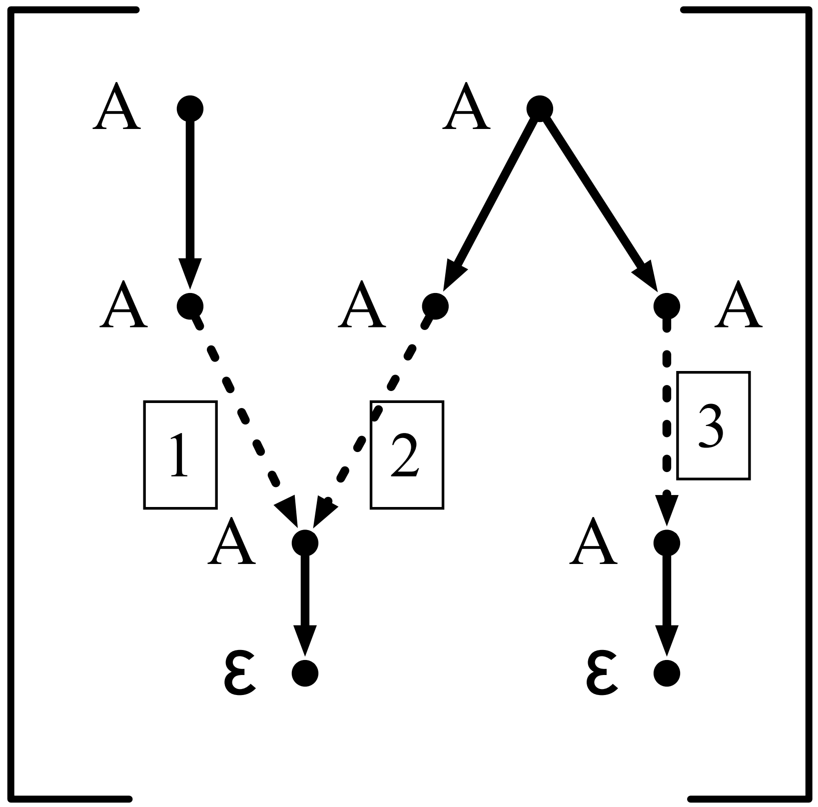
The UVG-DL parser

- Items: (A,i,j,M)
  here: = sets of dominance edges
- Axioms: incoming edges of terminal fragments
  Goal: empty set
- Inference rule:

$$[U_F] \ \frac{\mathsf{out}(h_1) \uplus S_1 \qquad \ldots \qquad \mathsf{out}(h_n) \uplus S_n}{\mathsf{in}(r) \cup S_1 \cup \ldots \cup S_n} \quad \left\{ \begin{array}{l} F \text{ is a fragment with root } r \\ \text{and holes } h_1, \ldots, h_n \end{array} \right.$$

- Intuition: Traverse graph bottom-up and keep track of sets of incoming dominance edges.

The UVG-DL parser in action



{1,2}

{3}

{2}

{1}

{}

$$[U_F] \; \frac{\mathsf{out}(h_1) \uplus S_1 \qquad \ldots \qquad \mathsf{out}(h_n) \uplus S_n}{\mathsf{in}(r) \cup S_1 \cup \ldots \cup S_n} \quad \begin{cases} F \text{ is a fragment with root } r \\ \text{and holes } h_1, \ldots, h_n \end{cases}$$

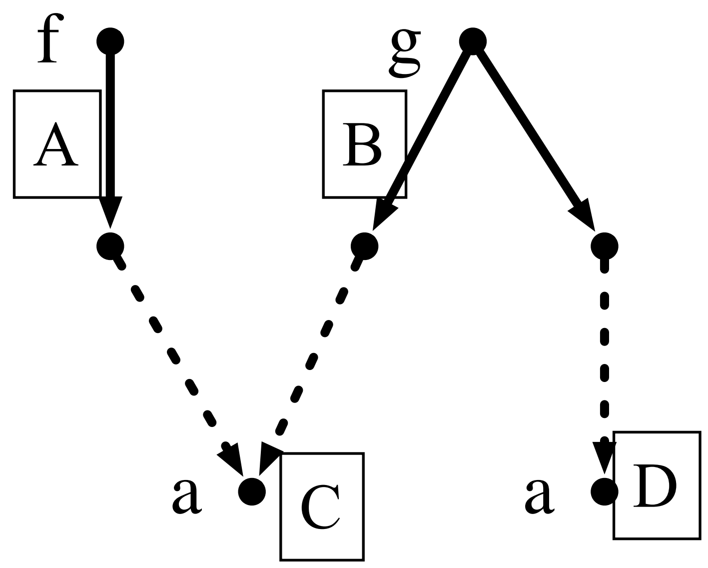
The dominance graph solver

- Items: subgraphs of the whole dominance graph
- Axiom: complete graph, then saturate with rule:

$$[D_F] \ \frac{G_0}{G_1 \quad \dots \quad G_n} \quad \left\{ \begin{array}{l} G_0 \text{ is connected, } F \text{ is a free fragment in} \\ G_0\text{, and the weakly connected components} \\ \text{of } G_0 - \{F\} \text{ are } G_1, \dots, G_n \end{array} \right.$$

- If we never encounter a non-free subgraph, this process terminates with one-fragment subgraphs.
- Enumerate configurations by applying rule bottom-up with axioms $\{F_1\}, \dots \{F_n\}$, goal item G.

The graph solver in action



{A,B,C,D}

{B,C,D}

{A,C}

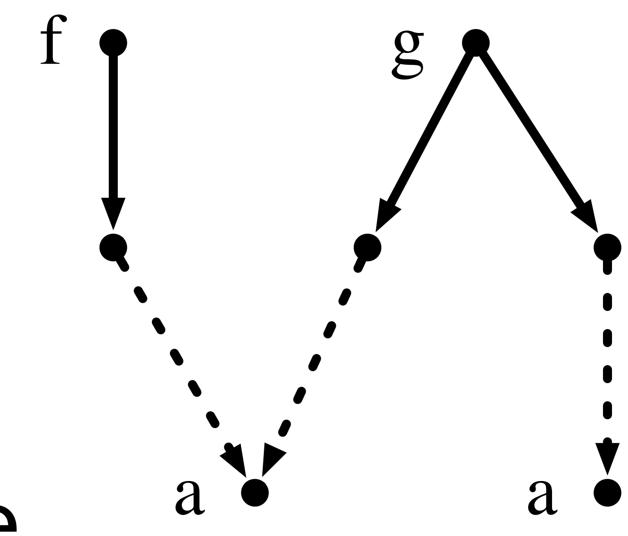
{D}

{C}

$$[D_F]\ \frac{G_0}{G_1 \quad \dots \quad G_n}$$

G_0 is connected, F is a free fragment in G_0, and the weakly connected components of $G_0 - \{F\}$ are $G_1, \dots, G_n$

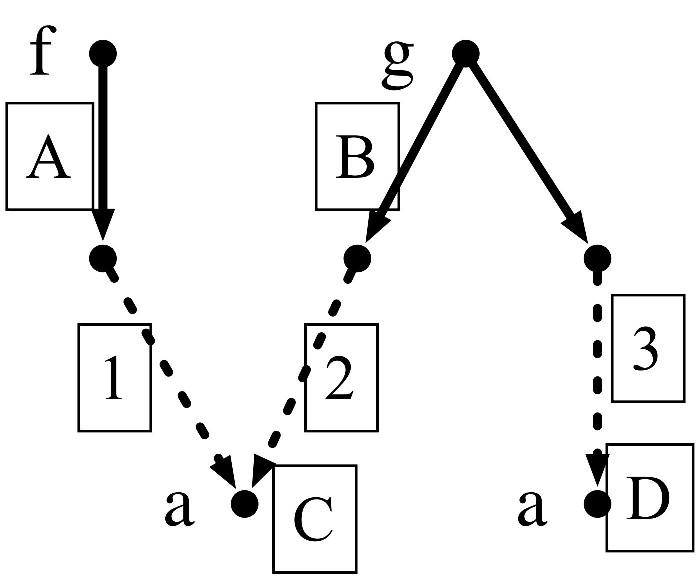
Cuts and cut subgraphs

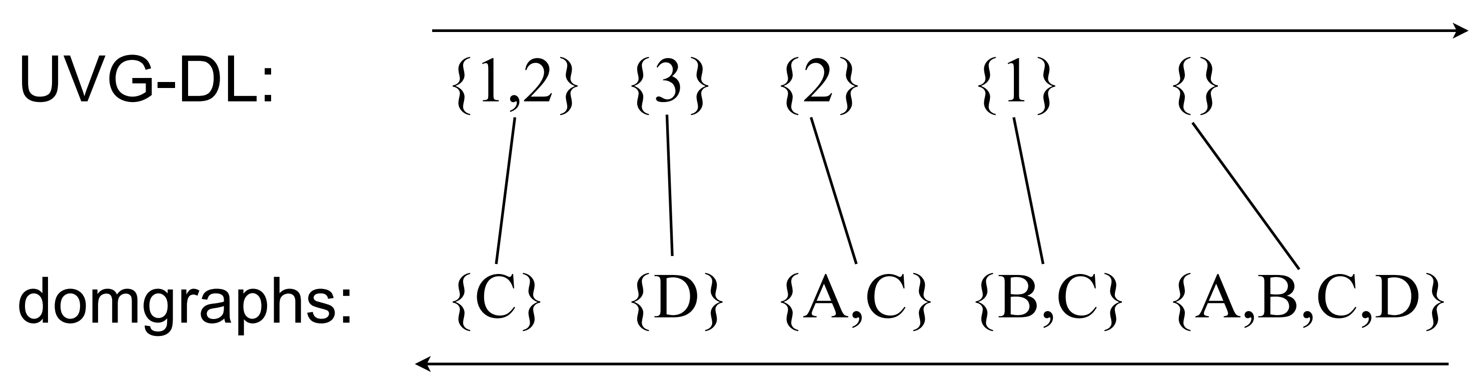


- Identify subgraphs uniquely by the set of dominance edges into them.
- Call set D' = {e_1,...,e_n} of dominance edges a cut iff
  - source and target of each e_i disconnected in G-D'
  - all targets pairwise connected in G-D'
- For each cut D', there is a unique subgraph that is connected, downwards closed, and has incoming edges D'. (And vice versa.)
  Call such graphs cut subgraphs.

Relating the charts

- The chart of a configurable, hnc dominance graph G contains exactly the configurable cut subgraphs of G.
- The chart of ε wrt the UVG-DL encoding of a configurable, hnc dominance graph G contains exactly the configurable cuts of G.
- Consequence: The charts are isomorphic.





Conclusion

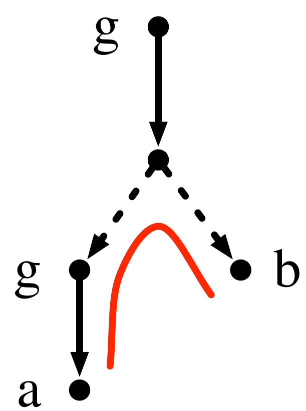
- Independently developed dominance-based tree description formalisms for syntax and semantics.
- Answered long-standing question: Can encode dominance graphs into UVG-DL.
- For configurable, hypernormally connected dominance graphs, the standard algorithms compute the same charts.
- Transfer theory and algorithms between formalisms.

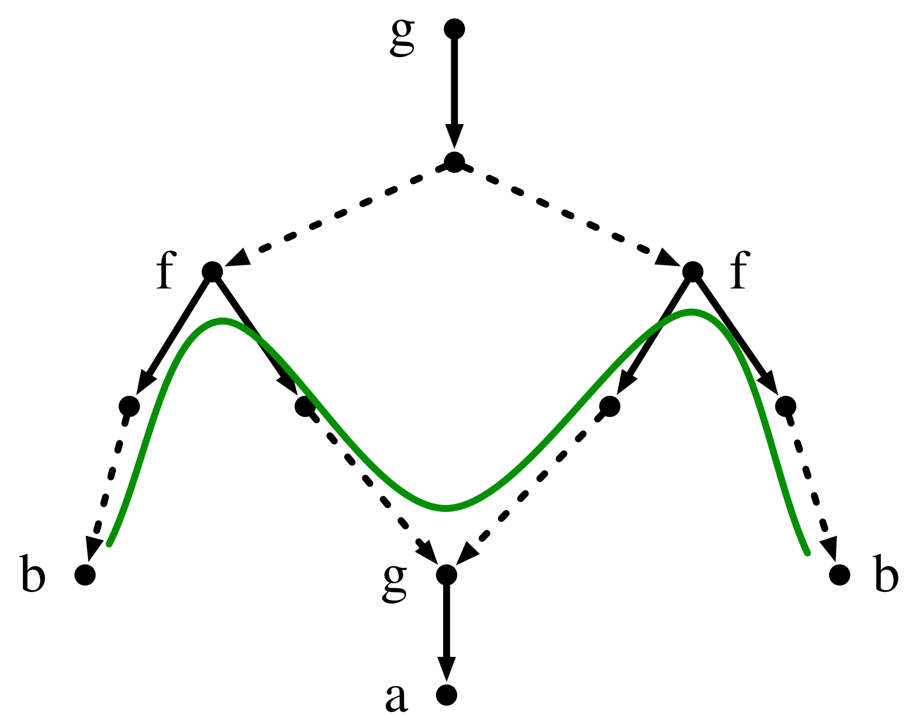
Future work

- More transfer of theory:
  - UVG-DL as natural extension path for dominance graphs
  - dominance graphs as benign fragment of UVG-DL
- Investigate division of labor between syntax and semantics:
  - scope ambiguities in syntax or semantics?
  - integrate scope underspecification methods into purely UVG-DL-driven parsers?

Hypernormal paths

- Hypernormal path: an undirected path in a dominance graph that doesn't use two dominance edges out of the same hole.





- Dominance graph is hypernormally connected (or hnc, or a net) iff every pair of nodes is connected by a hypernormal path.