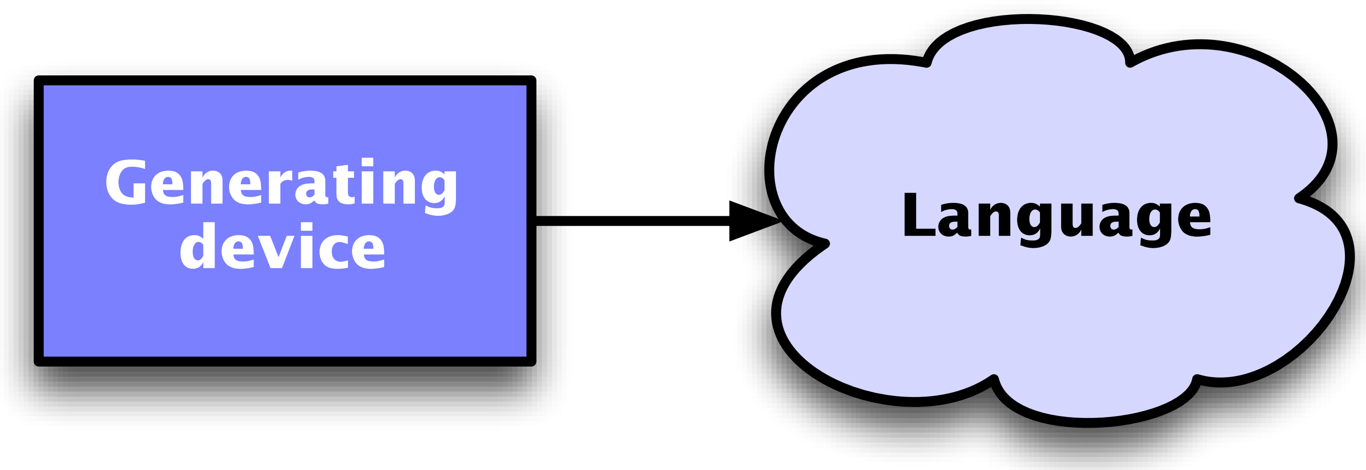
About Well-Nested Drawings

Marco Kuhlmann
Programming Systems Lab
Saarland University

January 27, 2005

IGK Colloquium, Saarbrücken

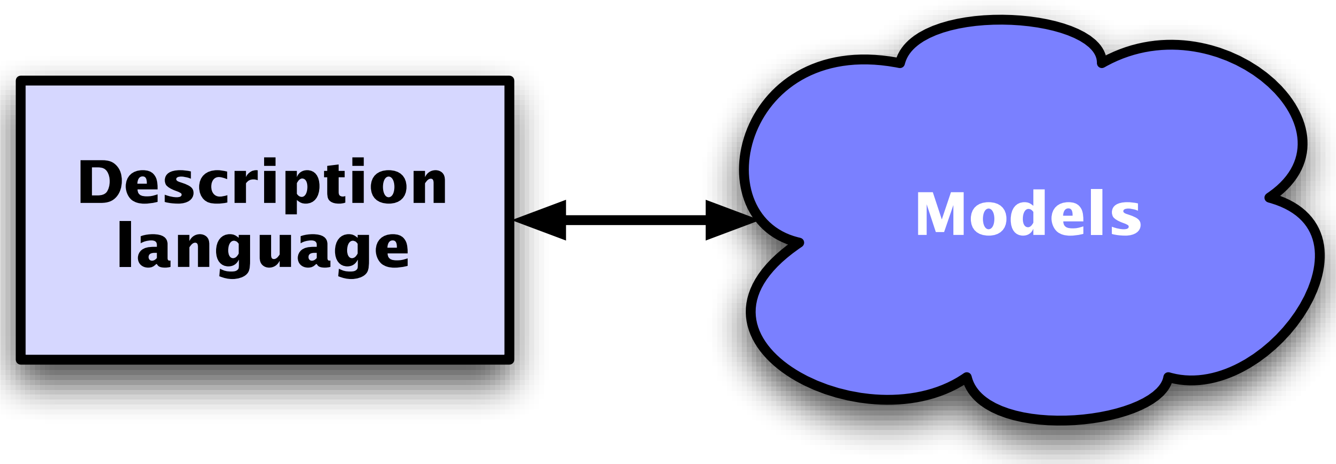
The bigger picture



Generative frameworks

- abstract generating device (G)
- language as the output of that device
- structure S
- S well-formed, if it is generated by G

- **linguistic structures: *a posteriori***



Model-theoretic frameworks

- class of models (M)
- description languages to talk about models
- structure S
- S well-formed, if its description is satisfied in M

- **linguistic structures: *a priori***

Benefits of model-theoretic approaches

- **partial and ambiguous information**
  - underspecified representations
  - syntax/semantics interface (DDKST @ COLING 2004)
- **modelling and methodology**
  - *a priori* notion of linguistic structures
  - choice among different description languages

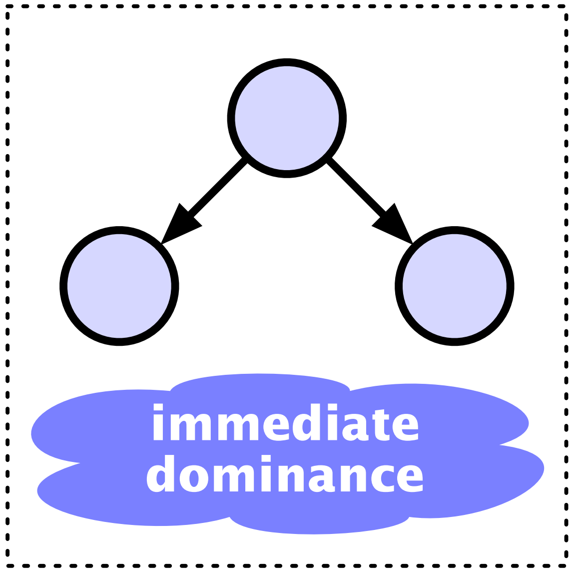
Questions

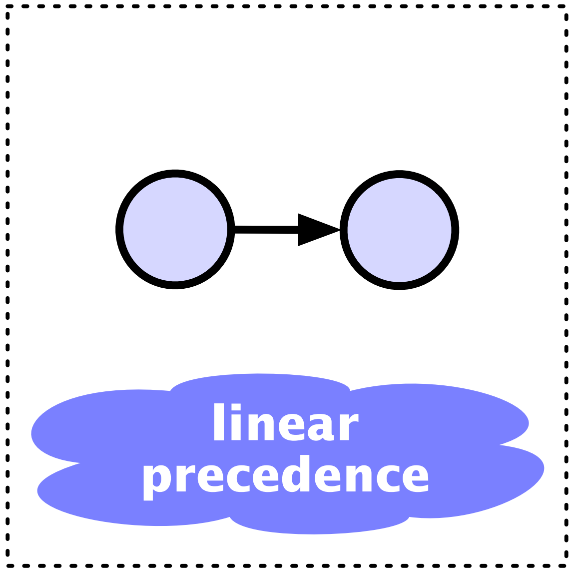
- **What class of structures should we consider?**
- What languages should we use to talk about it?

This talk

- The bigger picture
- **Drawings with gaps**
- Well-nested drawings
- Towards an algorithmic characterisation
- Future work

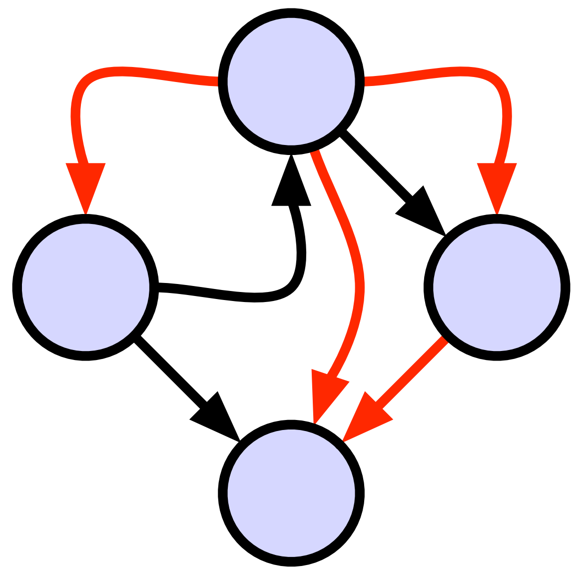
Drawings with gaps





Two dimensions

- **vertical dimension**
  - constituency
  - dependency
- **horizontal dimension**
  - word order
  - discontinuity

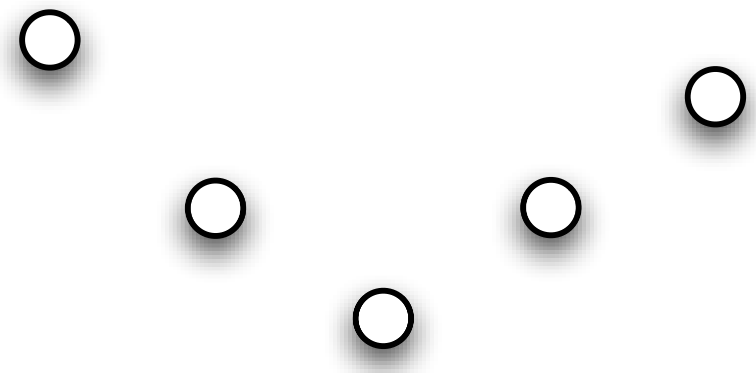


Relational structures

- **ingredients**
  - (non-empty) set of nodes
  - binary relations on the nodes
- **examples**
  - trees (ordered or unordered)
  - feature structures

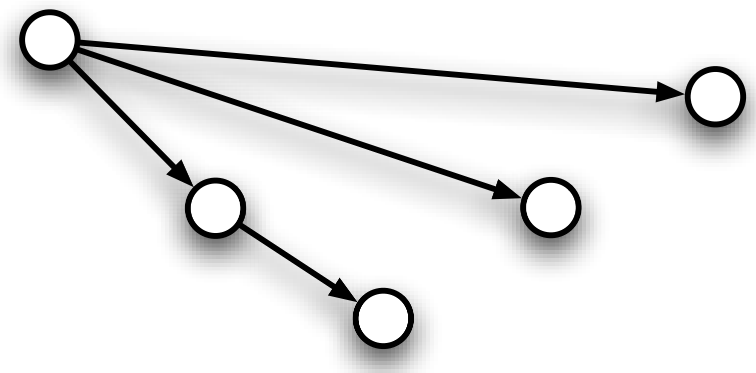
Drawings

- relational structures with two relations



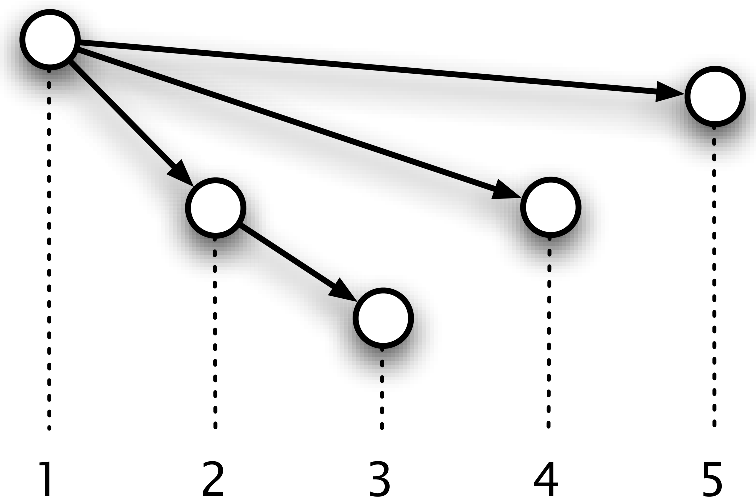
Drawings

- relational structures with two relations
- (finite) set of nodes



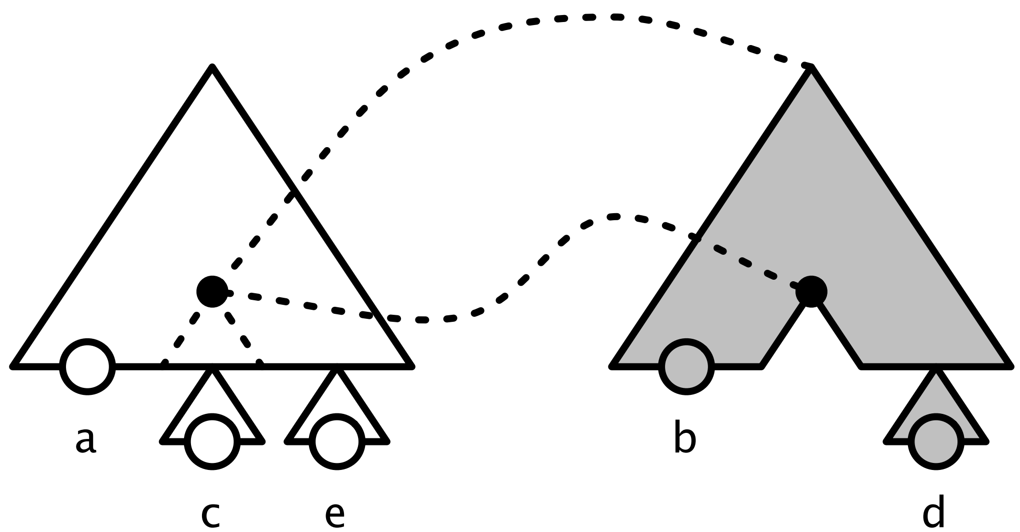
Drawings

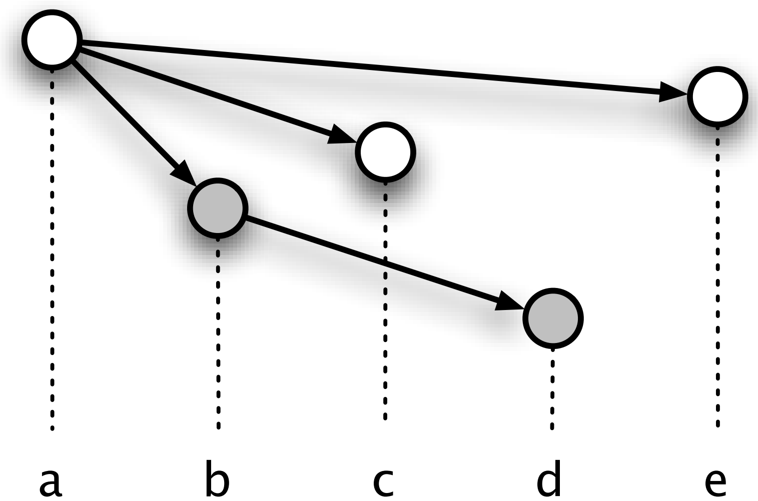
- relational structures with two relations
- (finite) set of nodes
- rooted tree S (successorship)



Drawings

- relational structures with two relations
- (finite) set of nodes
- rooted tree S (successorship)
- linear order P (precedence)





Drawings for TAG

- strongly lexicalised TAG
- nodes in the drawing: anchors
- tree: *derivation tree*
- order: order of anchors in the *derived tree*

- **Adjunction may cause ‘crossing edges’!**

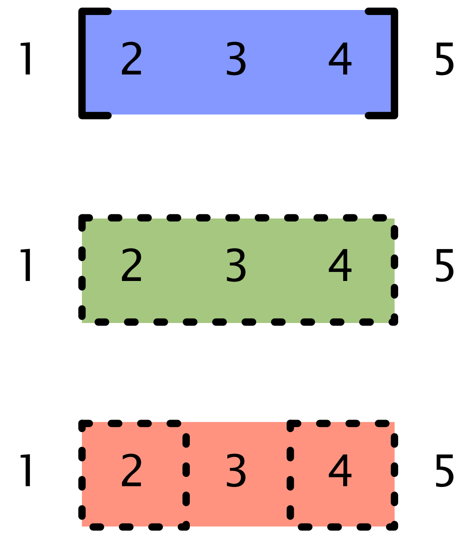
Convex sets and gaps

1 2 3 4 5

$<$

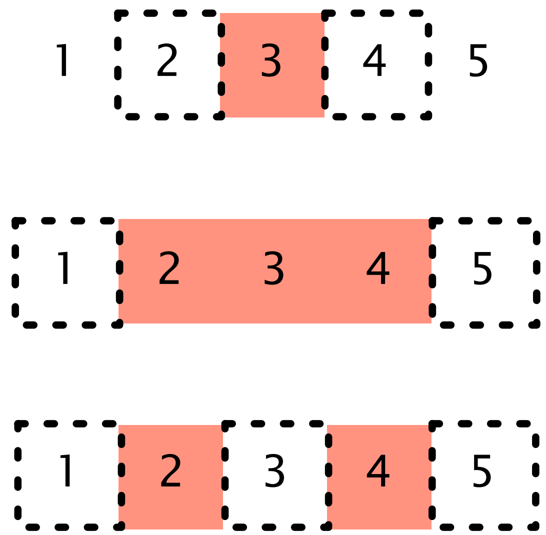
Strict linear orders

- Pair of a **set S** …
- … and a **binary relation R over S** that is
  - irreflexive,
  - transitive, and
  - trichotomic.



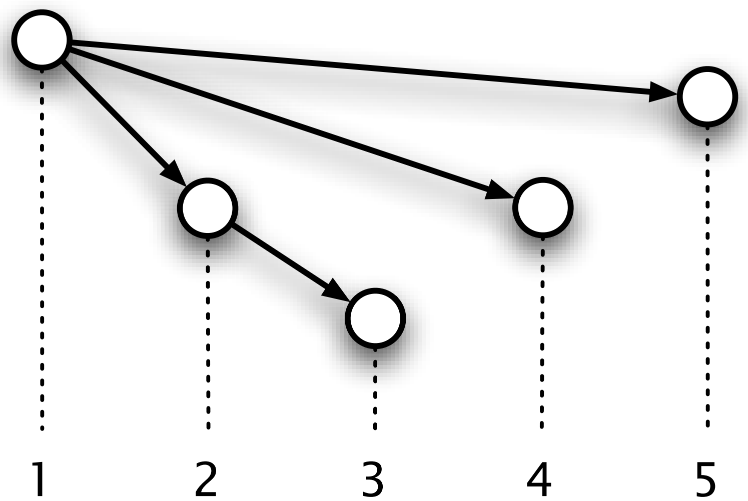
Convex sets

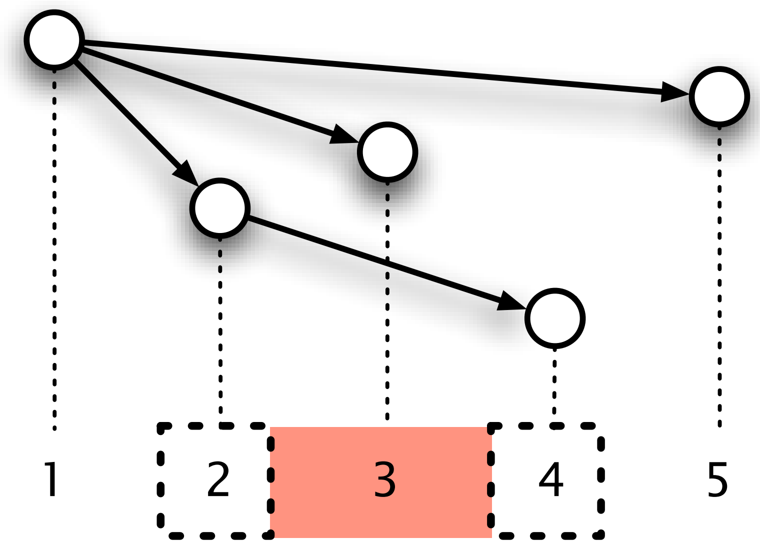
- **interval** [*a*,*b*]
  - contains all elements *x* such that $a \leq x \wedge x \leq b$
  - *a* and *b* are the *endpoints* of the interval
- **convex hull** of a set S
  - smallest interval that contains S
  - sets S such that S = H(S) are *convex*



Gaps

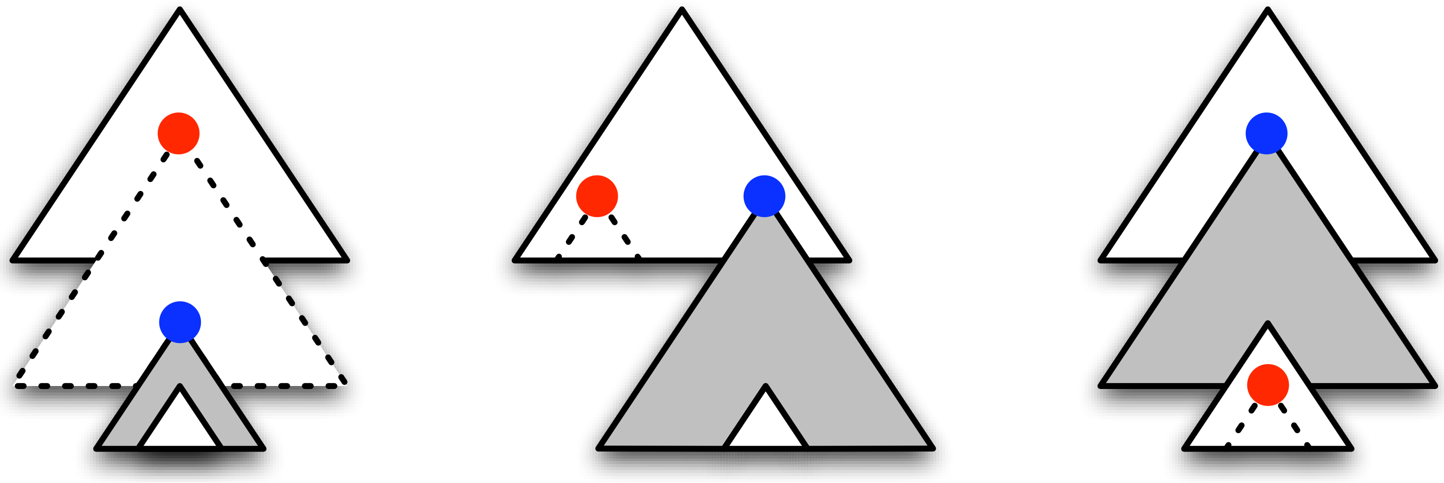
maximal intervals in the ‘holes’ of a set
(with respect to the strict linear order)





Drawings with gaps

- gap in a drawing =
  gap in the yield of one of its nodes
- drawings without gaps are *projective*
- gap degree of a drawing =
  maximal number of gaps for one of its nodes

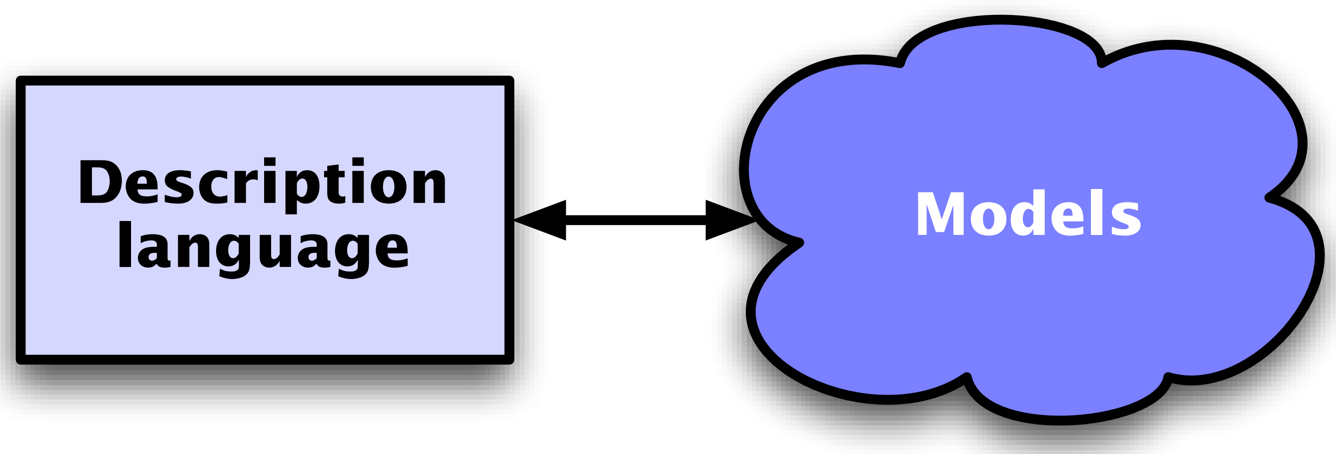


TAG is gap 1

- **adjunction creates gaps**
- **additional adjunctions**
  - gaps are inherited
  - new (disjoint) gaps are created
  - gaps are extended

Previous work

- **generative approach**
  - dependency trees with gaps (Platek et. al.)
  - linear specification language (Penn, Suhre)
- **model-theoretic approach**
  - pseudo-projectivity (Kahane et. al.)
  - multiplanarity (Yli-Jyrä)



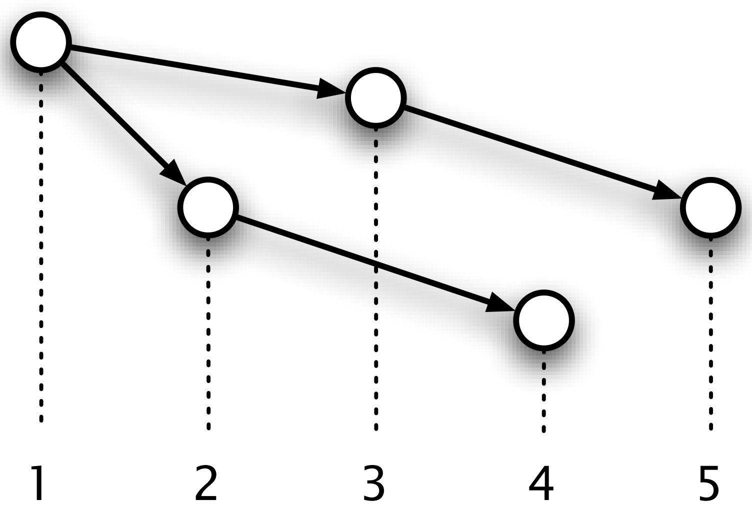
Model-theoretic frameworks

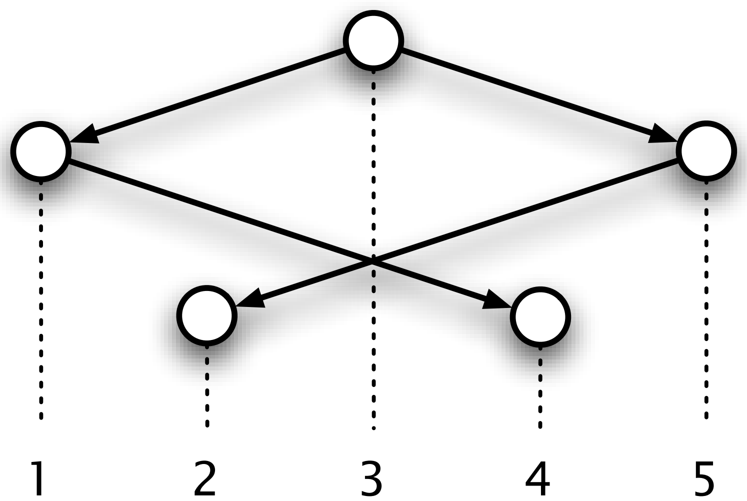
- **two alternatives**
  - stronger models
  - expressive description language
- **go for the former**
  - models should capture linguistic intuition
  - efficient algorithms

This talk

- The bigger picture
- Drawings with gaps
- **Well-nested drawings**
- Towards an algorithmic characterisation
- Future work

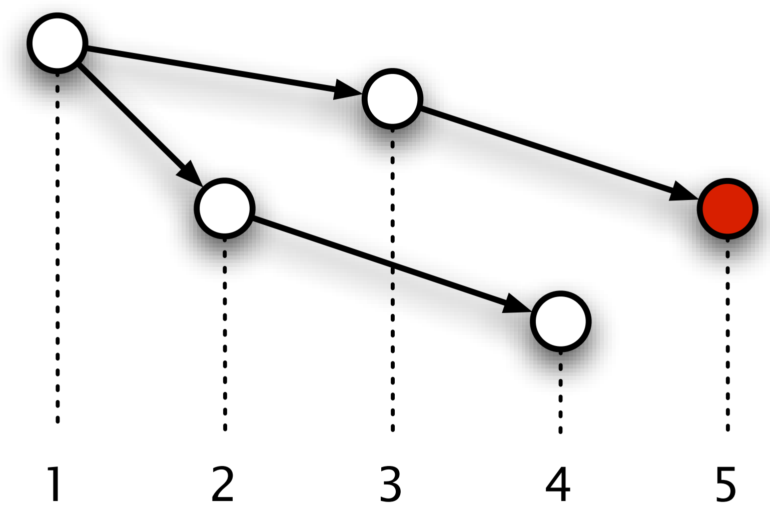
Well-nested drawings

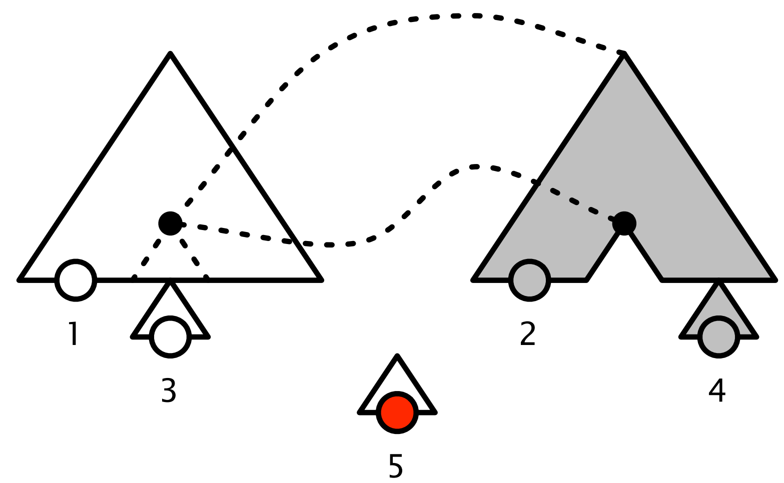




Observation

Not all gap-1 drawings are produced by a TAG.





One way and the other

- **In TAG, gaps are closed downwards:**
  - node 3 is in a gap in the yield of node 2
  - but its child (node 5) is not
- **In TAG, gaps are closed upwards:**
  - node 4 is in a gap in the yield of node 3
  - but its parent (node 2) is not

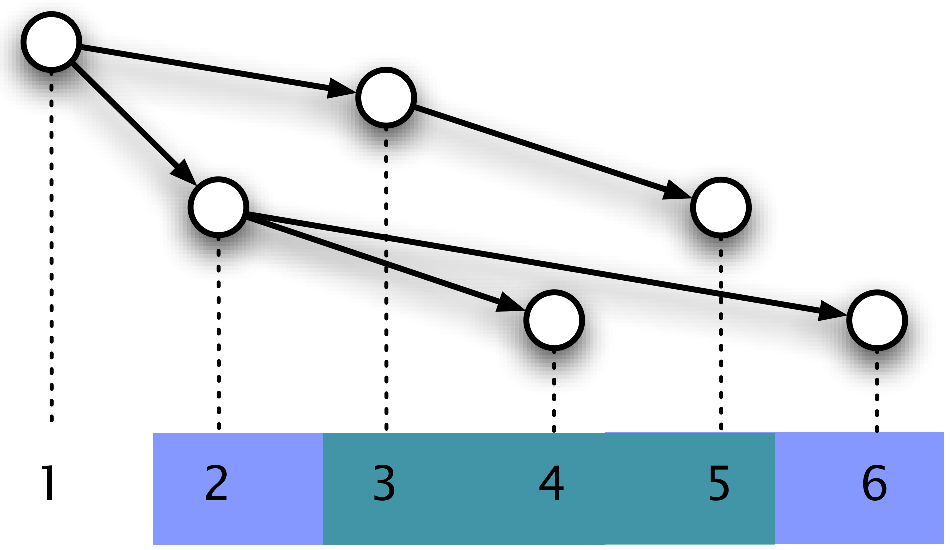
Well-nestedness

- **intuition**
  - well-nested: ‘obtainable by adjunctions’
  - extends to drawings with gap degree > 1
- **drawings for TAG**
  - well-nested and gap 1
  - necessary and sufficient

$$\forall\, u, v \in V: \quad C(u) = C(v) \;\vee\; C(u) \subset C(v) \;\vee\; C(u) \supset C(v) \;\vee\; C(u) \perp C(v)$$

Formalising well-nestedness

- **The arboreal tesseratomy** …
  - four relations between nodes in a tree
  - equality, (inverse) dominance, disjointness
- … **should extend to drawings.**
  - cover = convex hull of the yield of a node
  - four relations between covers in a drawing



Fishy things

- **Non-monotonic behaviour**
  - The definition only looks at the covers.
  - It cannot 'distinguish' between different gaps.
  - Result: Drawings that are not well-nested can be 'repaired' by introducing new gaps.
- **We do not want this to happen!**

$$\forall\, u, v \in V: \quad C(u) = C(v) \;\vee\; C(u) \subset C(v) \;\vee\; C(u) \supset C(v) \;\vee\; C(u) \perp C(v)$$

Solution

- A drawing is well-nested if and only if the covers of the nodes in all subsets of V form a tesseratomic family

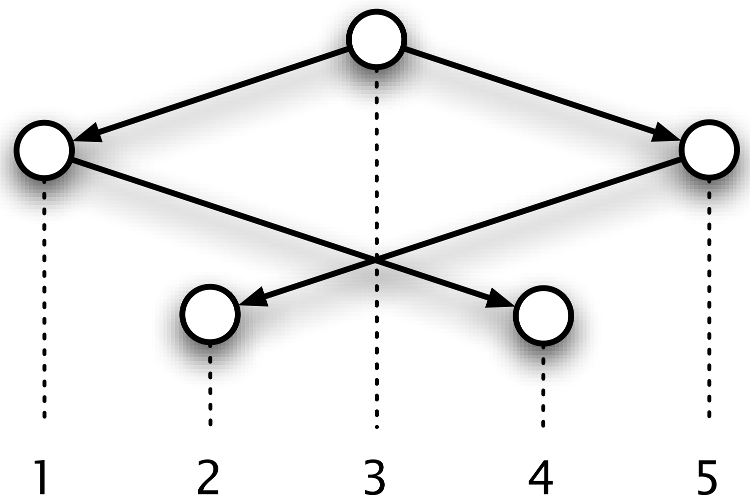
An algorithmic characterisation

The goal

An algorithm that tests
whether or not a given description
can be interpreted as a well-nested drawing.

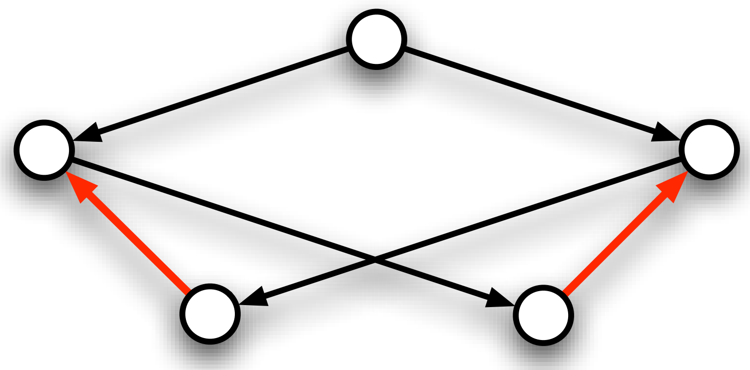
Two sides of the same coin

- **Relational structures offer two perspectives**
  - set theory: elements and relations
  - graph theory: nodes and edges
- **That’s nice for algorithms!**



Gap graphs

- **Rationale: ‘Making gaps explicit.’**
- graph on the same node set
- contains all the tree edges from the drawing
- contains additional ‘gap edges’

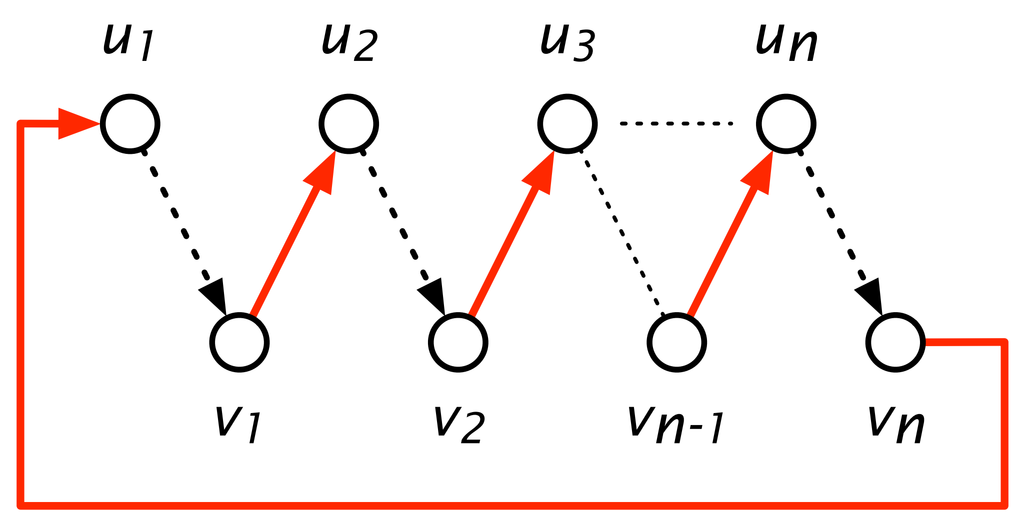


Gap graphs

- **Rationale: ‘Making gaps explicit.’**
- graph on the same node set
- contains all the tree edges from the drawing
- contains additional ‘gap edges’

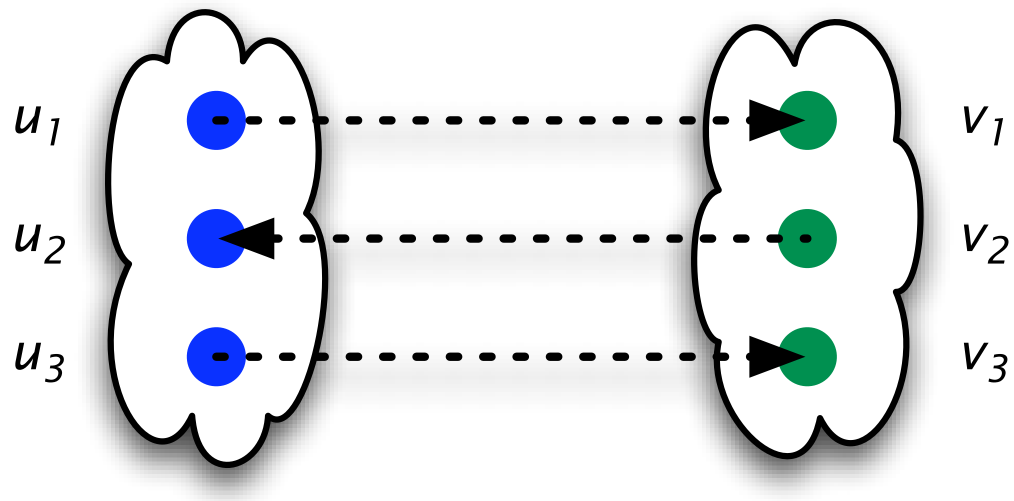
Stating the non-obvious

A drawing is well-nested if and only if its gap graph is acyclic.



Part of the proof

- Assume that the drawing is well-nested.
- If the gap graph contains a cycle,
  it contains a cycle in normal form.
- Each path *u*…*vu’* in the cycle
  translates into the requirement that
  C(*u*) is properly included in **C**(*u’*).
- Thus, **C**(u_1) should be properly included in **C**(u_1).



Well-nestedness, put differently

- two components
- connected by dominance edges
- an alternating path with precedence edges
- **cannot be well-nested!**

Future work

- complete all proofs
  (joint work with M. Möhl and R. Grabowski)
- design the algorithm
  (joint work with M. Bodirsky)
- think about the description language to use
- linguistic grounding
- look at other grammar formalisms