# Unterspezifikation in der Semantik
# Normal Dominance Constraints

Laura Kallmeyer
Heinrich-Heine-Universität Düsseldorf

Wintersemester 2011/2012





**Overview**

1. Idee
2. Dominance Constraints
3. Normal Dominance Constraints
4. Satisfiability and Constraint Solving

[Althaus et al., 2003, Koller et al., 1998, Koller et al., 2000]





**Idee (1)**

Beobachtung:

- Logische Ausdrücke haben eine baumförmige snytaktische Struktur.
- Unterspezifizierung bedeutet, dass Teilbäume schon komplett bekannt sind und dass wir außerdem Dominanzbeziehungen zwischen den Teilbäumen kennen.

Relationen auf Knoten in Bäumen:

- Unmittelbare Dominanz (*immediate dominance*) ist die Mutter-Tochter Relationen auf den Knoten eines Baums.
- Dominanz (*dominance*) ist deren reflexive transitive Hülle.





**Idee (2)**

(1) everybody contradicts something

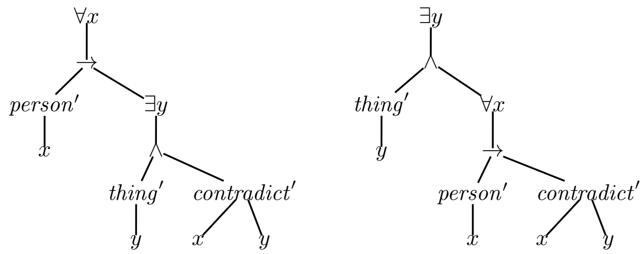
Die beiden Lesarten (in Prädikatenlogik, mit Quantoren ∀, ∃):

1. $\forall x(\mathit{person}'(x) \rightarrow \exists y(\mathit{thing}'(y) \wedge \mathit{contradict}'(x,y)))$
2. $\exists y(\mathit{thing}'(y) \wedge \forall x(\mathit{person}'(x) \rightarrow \mathit{contradict}'(x,y)))$

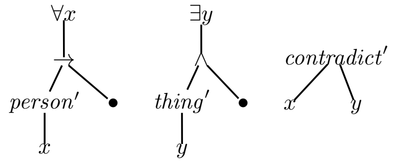
Schreibt man die Operatoren → und ∧ in Präfixnotation, so bekommt man

1. $\forall x(\rightarrow(\mathit{person}'(x), \exists y(\wedge(\mathit{thing}'(y), \mathit{contradict}'(x,y)))))$
2. $\exists y(\wedge(\mathit{thing}'(y), \forall x(\rightarrow(\mathit{person}'(x), \mathit{contradict}'(x,y)))))$

**Idee (3)**




The two trees have three common subtrees:

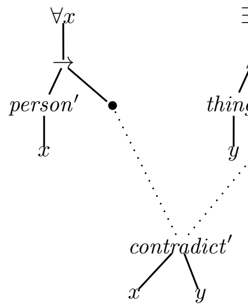







**Idee (4)**

Es gibt folgende Dominanzbeziehungen (dargestellt durch gepunktete Kanten)



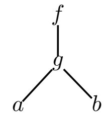





**Dominance Constraints (1)**

Terme werden also durch entsprechende Bäume beschrieben:

Term $f(g(a, b))$ $\leadsto$




Die Knotenlabels sind Funktionssymbole aus einem Alphabet $\Sigma$:

$\Sigma$ ist ein Alphabet von Funktionssymbole, wobei jedes $f \in \Sigma$ eine eindeutige Stelligkeit $ar(f) \geq 0$ hat. (Stelligkeit 0 bedeutet, dass es sich um eine Konstante wie z.B. $a$ in obigem Baum handelt.)





**Dominance Constraints (2)**

Ein Baum ist ein endlicher gerichteter Graph $\langle V, E\rangle$, wobei

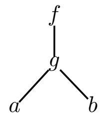
- $V$ eine endliche Menge von Knoten ist, und
- $E \subseteq V \times V$ die Menge von Kanten. Es gilt,
  - dass der Eingangsgrad jedes knoten n $V$ höchstens 1 ist und
  - dass es genau einen Wurzelknoten (Eingangsgrad 0) gibt.

Ein Konstruktorbaum ist ein Paar $\langle T, L\rangle$, wobei $T = \langle V, E\rangle$ ein Baum ist und $L$ eine Labelfunktion, die

- jedem Knoten $v \in V$ ein Label aus $\Sigma$ zuordnet, und
- jeder Kante $e \in E$ eine natürliche Zahl zuordnet. Dabei muss gelten: Für jedes $v \in V$ und alle $k$ mit $1 \leq k \leq ar(L(v))$ gibt es genau ein $u \in V$, so dass $L(\langle v, u\rangle) = k$.

($L(\langle v, u\rangle) = k$ bedeutet "Kante zur $k$ten Tochter".)

### Dominance Constraints (3)




$\langle\langle V, E\rangle, L\rangle$ mit

- $\Sigma = \{f, g, a, b\}$, $ar(f) = 1, ar(g) = 2, ar(a) = 0, ar(b) = 0$,
- $V = \{v_1, v_2, v_3, v_4\}$,
- $E = \{\langle v_1, v_2\rangle, \langle v_2, v_3\rangle, \langle v_2, v_4\rangle\}$,
- $L(v_1) = f, L(v_2) = g, L(v_3) = a, L(v_4) = b$ und
- $L(\langle v_1, v_2\rangle) = 1,\ L(\langle v_2, v_3\rangle) = 1,\ L(\langle v_2, v_4\rangle) = 2$





### Dominance Constraints (4)

Zu einem Konstruktorbaum $\tau = \langle\langle V, E\rangle, L\rangle$, der ja Dominanz nur implizit zur Verfügung stellt, gibt es eine Baumstruktur $\mathcal{M}^\tau = \langle V, \lhd^{*\tau}, \hat{f}_1, \dots, \hat{f}_n\rangle$, wobei

- $\lhd^{*\tau}$ die Dominanrelation ist, d.h. $v \lhd^{*\tau} u$ gdw. es in $E$ einen Pfad von $v$ nach $u$ gibt.
- $\hat{f}_1, \dots, \hat{f}_n$ Relationen für alle Elemente aus $\Sigma = \{f_1, \dots, f_n\}$ sind. Diese Relationen sind wie folgt definiert:
  $\hat{f}$ ist $ar(f) + 1$-stellig, d.h., $\hat{f} \subset V^{ar(f)+1}$, und
  $\langle u, v_1, \dots, v_{ar(f)}\rangle \in \hat{f}$ gdw. $L(u) = f$ und $L(u, v_i) = i$ für alle $1 \leq i \leq ar(f)$.

In diesem Baumstrukturen sind genau die Sachen explizit, über die wir in unterspezifizierten Repräsentationen reden möchten: Terme bestehend aus Prädikaten $\hat{f}$ und deren Argumenten und Dominanzbeziehungen zwischen Termen.





### Dominance Constraints (5)

Jetzt definieren wird Dominanzconstraints mit einer modelltheoretischen Semantik, wobei Baumstrukturen die Modelle sind:

*Vars* sei ein Alphabet von Variablen.

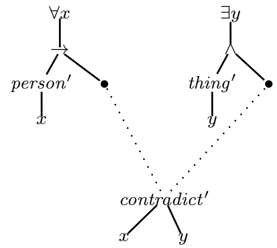
- Für alle $X, Y \in \mathit{Vars}$ ist $X \lhd^* Y$ ein Dominanzconstraint.
- Für alle $X, Y \in \mathit{Vars}$ ist $X \neq Y$ ein Dominanzconstraint.
- Für alle $f \in \Sigma$ und $X, X_1, \dots, X_{ar(f)} \in \mathit{Vars}$ ist $X : f(X_1, \dots, X_{ar(f)})$ ein Dominanzconstraint.
- Für alle Dominanzconstraints $\varphi_1, \varphi_2$ ist $\varphi_1 \wedge \varphi_2$ eine Dominanzconstraint.
- Das sind alle Dominanzconstraints (über $\Sigma$).





### Dominance Constraints (6)

Bsp.:




$X_1 : \forall x(X_2) \wedge X_2 :\rightarrow (X_3, X_4) \wedge X_3 : \mathit{person}'(X_5) \wedge X_5 : x() \wedge$
$X_6 : \exists y(X_7) \wedge X_7 : \wedge(X_8, X_9) \wedge X_8 : \mathit{thing}'(X_{10}) \wedge X_{10} : y() \wedge$
$X_{11} : \mathit{contradict}'(X_{12}, X_{13}) \wedge X_{12} : x() \wedge X_{13} : y() \wedge$
$X_4 \lhd^* X_{11} \wedge X_9 \lhd^* X_{11}$

### Dominance Constraints (7)

Jetzt müssen wir noch die Semantik von Dominance Constraints definieren:

$\varphi$ sei ein Dominanconstraint mit Variablen $Var(\varphi)$. Eine Baumstruktur $\mathcal{M}\tau$(bzgl. $\Sigma$) mit Knotenmenge $V$ erfüllt $\varphi$ unter einer Variablenzuweisung $\alpha : Var(\varphi) \to V$ wenn jedes Konjunkt aus $\varphi$ erfüllt ist.

- $X_1 \lhd^* X_2$ ist erfüllt gdw. $\alpha(X_1) \lhd^{*\tau} \alpha(X_2)$.
- $X : f(X_1, \ldots, X_{ar(f)})$ ist erfüllt gdw. $\langle \alpha(X), \alpha(X_1), \ldots, \alpha(X_{ar(f)}) \rangle \in \hat{f}$.





### Dominance Constraints (8)

Gegeben ein Dominanzconstraint gibt es zwei Fragen, die man entscheiden können möchte:

1. **Erfüllbarkeit**: Gibt es ein Modell (i.e., eine Baumstruktur) und eine Variablenzuweisung, bzgl. der das Dominanzconstraint das Modell erfüllt?
2. **Lösungen Aufzählen**: Welche Modelle gibt es für ein gegebenes Dominanzconstraint?

Für Dominanzconstraints ist das Erfüllbarkeitsproblem NP-vollständig [Koller et al., 1998] (Problem: Fragmente können zu sehr überlappen).

Frage: Gibt es nützliche Untergruppen von Dominanzconstraints, für die das Erfüllbarkeitsproblem polynomiell ist?





### Normal Dominance Constraints (1)

Ein Dominanzconstraint $\varphi$ ist normal gdw. für alle $X, Y, Z \in Var(\varphi)$:

1. $X \neq Y$ in $\varphi$ gdw. es Konjunkte $X : f(\ldots)$ und $Y : g(\ldots)$ in $\varphi$ gibt (kein overlap);
2. $X$ kommt höchstens einmal als Mutter und höchstens einmal als Tochter in Konjunkten der Form $X_1 : f(Y_1, \ldots, Y_{ar(f)})$ vor (Baumformat);
3. Wenn $X \lhd^* Y$ in $\varphi$, dann gibt es kein Konjunkt der Form $X : f(\ldots)$ für irgendein $f \in \Sigma$ und keine Konjunkt der Form $Z : f(\ldots Y \ldots)$ für irgendein $f \in \Sigma$ (Dominanzen gehen von Holes zu Labels).
4. Wenn $X \lhd^* Y$ in $\varphi$, dann gibt es $Z \in Var(\varphi)$ und $f \in \Sigma$, so dass $Z : f(\ldots X \ldots)$ ein Konjunkt in $\varphi$ ist (keine leeren Fragmente).





### Normal Dominance Constraints (2)

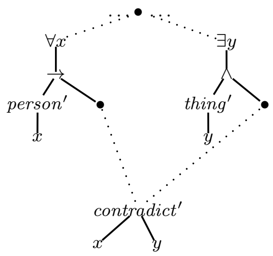
Theorem: Das Erfüllbarkeitsproblem für normale Dominanzconstraints ist in $\mathcal{O}((k+1)^3 n^2 \log n)$ lösbar, wobei $n$ die Anzahl Variablen im Constraint ist und $k$ die maximale Anzahl von Dominanzkanten in denselben Knoten.

$k$ ist in der Regel klein.

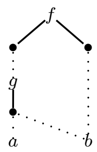
Um zu testen, ob ein Dominanzconstraint erfüllbar ist, benötigen wir den Begriff der Hypernormalität: Wir betrachten die graphische Darstellung eines Constraints. Ein Zyklus in der ungerichteten Version dieses Graphen heißt hypernormal , wenn er keine adjazenten Dominanzkanten enthält, die vom selben Knoten ausgehen.

## Normal Dominance Constraints (3)

Nicht hypernormale Zyklen können in der Regel aufgelöst werden.




Hypernormale Zyklen dagegen können nicht aufgelöst werden.








## Normal Dominance Constraints (4)

Es gilt folgende Proposition:

Ein normales Dominanzconstraint ist erfüllbar gdw. sein ungerichteter Constraintgraph keinen einfachen hypernormalen Zyklus enthält ("einfach" bedeutet, dass kein Knoten mehrfach durchlaufen wird).

Im Erfüllbarkeit zu testen, reicht es also, zu überprüfen, ob es einfache hypernormale Zyklen gibt.





## Normal Dominance Constraints (5)

Constraints Lösen und Lösungen aufzählen:

- Idee: Wir fügen so lange weitere Dominanzen hinzu, bis der Dominanzgraph baumförmig ist. Danach müssen nur noch die beiden Knoten einer Dominanzkante identifiziert werden.
- Genauer: Operation *Distr*:

  Aus

  $$\varphi \wedge X \lhd^* Z \wedge Y \lhd^* Z$$

  (d.h., $Z$ hat zwei eingehende Dominanzkanten) wird

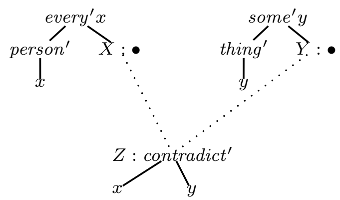
  $\varphi_1 = \varphi \wedge X \lhd^* R_\varphi(Y) \wedge Y \lhd^* Z$ oder $\varphi_2 = \varphi \wedge Y \lhd^* R_\varphi(X) \wedge X \lhd^* Z$

  wobei $R_\varphi(Y)$ ($R_\varphi(X)$) die Wurzelvariable des Terms mit Blatt $Y$ ($X$) ist.

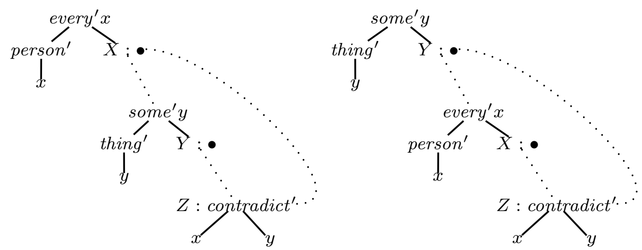




## Normal Dominance Constraints (6)




$\rightsquigarrow$



### Normal Dominance Constraints (7)

- Ein normales Dominanconstraint $\varphi$ heißt irredundant, wenn es kein normales Dominanzconsraint $\varphi'$ gibt, das wie $\varphi$ aussieht, nur weniger Dominanzbeziehungen enthält, und das $\varphi$ subsumiert. (D.h., Bäume, die $\varphi'$ erfüllen, erfüllen auch $\varphi$).
- Ein normales Dominanconstraint $\varphi'$ ist eine irredundante gelöste Form (*solved form*) von einem normalen Dominanzconstraint $\varphi$, wenn
  - $\varphi'$ alle Variablen, Labels und Ungleichheitsrelationen aus $\varphi$ enthält,
  - $\varphi'$ erfüllbar ist,
  - jedes Modell von $\varphi'$ auch ein Modell von $\varphi$ ist, und
  - der Graph von $\varphi'$ keinen Knoten mit zwei ausgehenden oder zwei eingehenden Dominanzkanten enthält.





### Normal Dominance Constraints (8)

Algorithmus zum Finden der irredundanten gelösten Formen eines Dominanzcontraints $\varphi$:

1. Überprüfe, ob $\varphi$ erfüllbar. Falls nicht, Abbruch.
2. Sonst: Mache $\varphi$ irredundant durch Entfernen überflüssiger Dominanzterme.
3. Falls $\varphi$ eine gelöste Form, Abbruch und Rückgabe von $\varphi$.
4. Sonst: Wende *Distr* auf $\varphi$ an und setze den Prozess mit den beiden neuen Dominazconstraints fort.





### Normal Dominance Constraints (9)

Es gilt [Koller et al., 2000]: Das Lösen normaler Dominanzconstraints ist polynomiell in der Größe des Constraints.

Genauer:

Dieser Algorithmus listet die irredundanten gelösten Formen eines normalen Dominanzconstraints $\varphi$ in einer Zeit von $\mathcal{O}((k+1)^4 n^4 N \log n)$, wobei $N$ die Anzahl der irredundanten gelösten Formen, $n$ die Anzahl Variablen in $\varphi$ und $k$ die maximale Anzahl eingehender Dominanzkanten pro Knoten im entsprechenden Dominanzconstraintgraph.

Vorsicht: $N$ kann immer noch exponentiell in der Größe von $\varphi$ sein.





### Normal Dominance Constraints (10)

Implementierung in utool
`http://www.coli.uni-saarland.de/projects/chorus/utool/`, the “Swiss Army Knife of Underspecification”, im Rahmen des CHORUS Projekts, Saarbrücken.

# References

[Althaus et al., 2003] Althaus, E., Duchier, D., Koller, A., Mehlhorn, K., Niehren, J., and Thiel, S. (2003). An efficient graph algorithm for dominance constraints. *Journal of Algorithms*, 48(1):194–219.

[Koller et al., 2000] Koller, A., Mehlhorn, K., and Niehren, J. (2000). A polynomial-time fragment of dominance constraints. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*.

[Koller et al., 1998] Koller, A., Niehren, J., and Treinen, R. (1998). Dominance Constraints: Algorithms and Complexity. In *Proceedings of the Third International Conference on Logical Aspects of Computational Linguistics (LACL '98)*, Grenoble, France.