Anwendungsbeispiel MAXIMUM TREE ORIENTATION

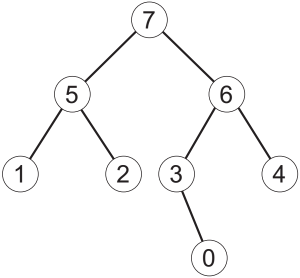
Dominikus Krüger

Universität Ulm

16. November 2011

Inhaltsverzeichnis

Maximum Tree Orientation

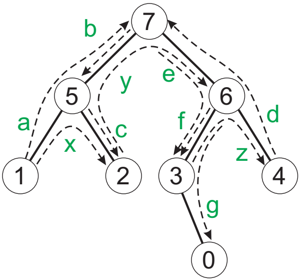


Definition (Maximum Tree Orientation)

Gegeben: Ein ungerichteter Baum $T = (V, E)$ und eine Menge $P \subseteq V \times V$ geordneter Paare von Knoten.

Gesucht: Eine Orientierung $\vec{T}$ von T, in der die größtmögliche Anzahl Paare $[v, w] \in P$ realisierbar ist, d. h. ein gerichteter Pfad von v nach w in $\vec{T}$ existiert.

Maximum Tree Orientation

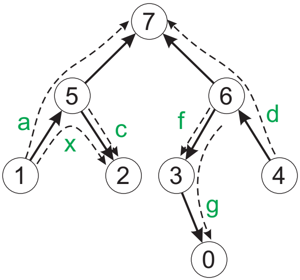


Definition (Maximum Tree Orientation)

Gegeben: Ein ungerichteter Baum $T = (V, E)$ und eine Menge $P \subseteq V \times V$ geordneter Paare von Knoten.

Gesucht: Eine Orientierung $\vec{T}$ von T, in der die größtmögliche Anzahl Paare $[v, w] \in P$ realisierbar ist, d. h. ein gerichteter Pfad von v nach w in $\vec{T}$ existiert.

Maximum Tree Orientation

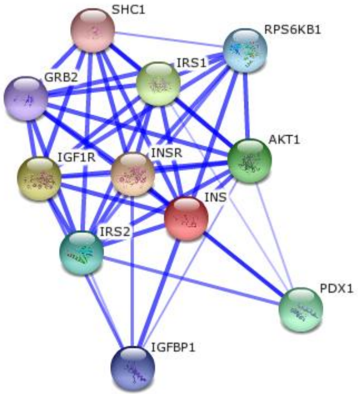


Definition (Maximum Tree Orientation)

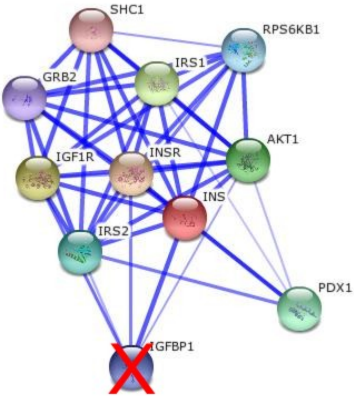
Gegeben: Ein ungerichteter Baum $T = (V, E)$ und eine Menge $P \subseteq V \times V$ geordneter Paare von Knoten.

Gesucht: Eine Orientierung $\vec{T}$ von T, in der die größtmögliche Anzahl Paare $[v, w] \in P$ realisierbar ist, d. h. ein gerichteter Pfad von v nach w in $\vec{T}$ existiert.

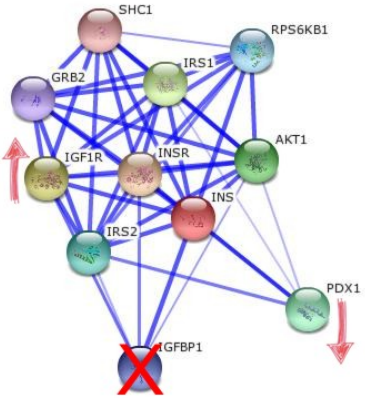
Motivation



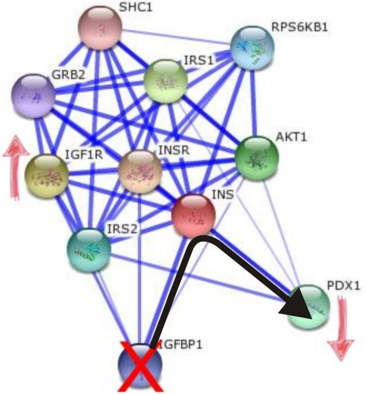
Motivation



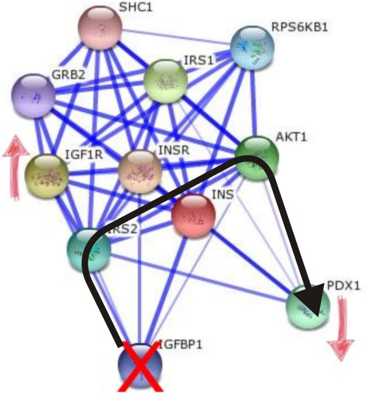
Motivation



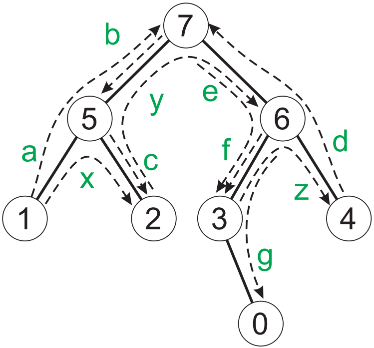
Motivation



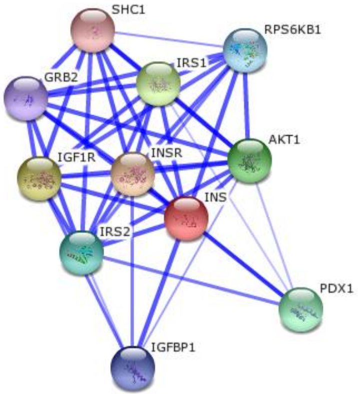
Motivation



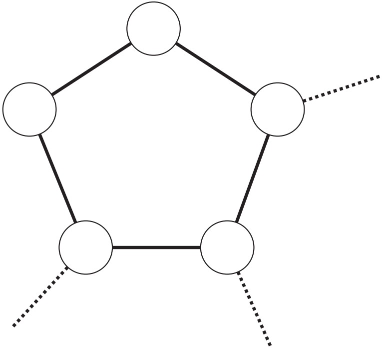
Motivation



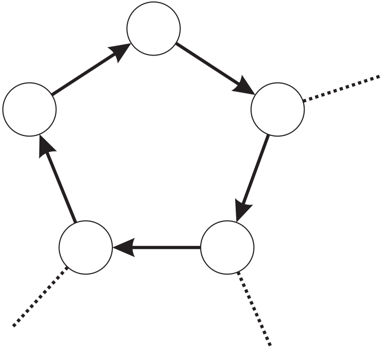
Motivation



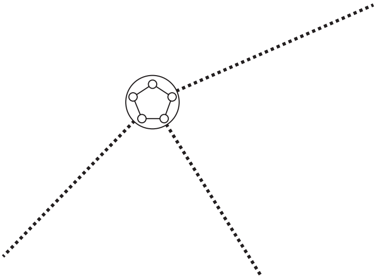
Warum ein Baum?



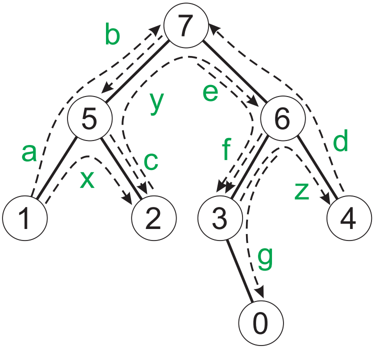
Warum ein Baum?



Warum ein Baum?



Warum ein Baum?



Integer Linear Programming

ILP

- Ganzzahliger Spezialfall linearer Optimierung
- Besteht aus:
  - zu optimierende Zielfunktion
  - einzuhaltende Nebenbedingungen
  - Variablen

Integer Linear Programming

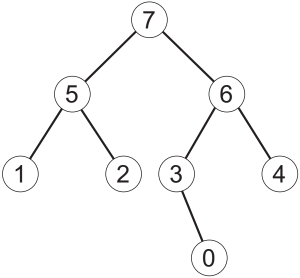
ILP

- Ganzzahliger Spezialfall linearer Optimierung
- Besteht aus:
  - zu optimierende Zielfunktion
  - einzuhaltende Nebenbedingungen
  - Variablen

Resultat von Lenstra

- ILP sind in polynomieller Zeit in der Anzahl der Variablen lösbar.
- ILP mit nur $f(k)$ Variablen
  $\rightarrow$ Problem ist fpt mit Parameter k

MTO Parameter: Anzahl Paare (1)

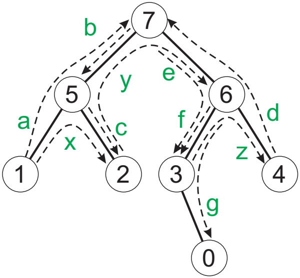


Integer Linear Programming

max:

Nebenbedingungen:

MTO Parameter: Anzahl Paare (1)

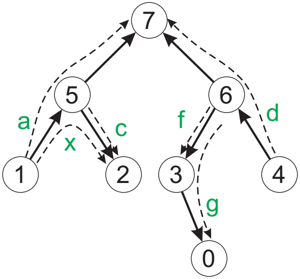


Integer Linear Programming

max:

Nebenbedingungen:

MTO Parameter: Anzahl Paare (1)

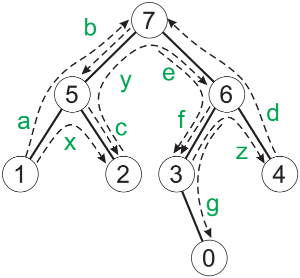


Integer Linear Programming

max:

Nebenbedingungen:

MTO Parameter: Anzahl Paare (1)

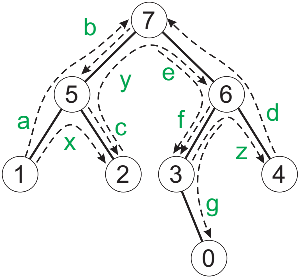


Integer Linear Programming

max:

Nebenbedingungen:

MTO Parameter: Anzahl Paare (1)

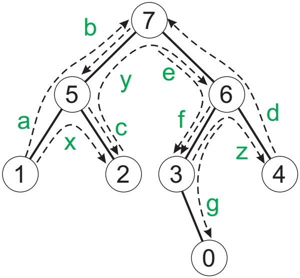


Integer Linear Programming

max:

Nebenbedingungen: $\forall p_i \in P : p_i \in \{1, 0\}$

MTO Parameter: Anzahl Paare (1)

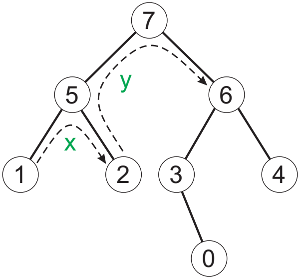


Integer Linear Programming

$$\max: \sum_{i=1}^{|P|} p_i$$

Nebenbedingungen: $\forall p_i \in P : p_i \in \{1, 0\}$

MTO Parameter: Anzahl Paare (1)

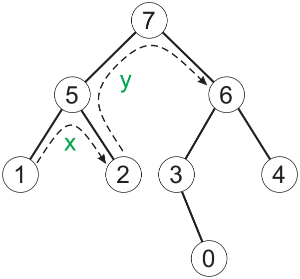


Integer Linear Programming

$$\text{max: } \sum_{i=1}^{|P|} p_i$$

Nebenbedingungen: $\forall p_i \in P : p_i \in \{1, 0\}$

MTO Parameter: Anzahl Paare (1)



Integer Linear Programming

$$\text{max: } \sum_{i=1}^{|P|} p_i$$

$$\text{Nebenbedingungen: } \forall p_i \in P : p_i \in \{1, 0\}$$

$$\forall p_i, p_j \in P, i \neq j, p_i \uparrow\downarrow p_j : p_i \cdot p_j = 0$$

Tiefenbeschränkter Suchbaum

Suchbaum

- Erschöpfende Suche
- Iterativ ebenenweiser Aufbau
- Pro Ebene Entscheidung für oder gegen mindestens ein Element
- Suchbaum maximal k Ebenen tief
  $\rightarrow$ maximal $2 \cdot 2^k$ Knoten

Tiefenbeschränkter Suchbaum

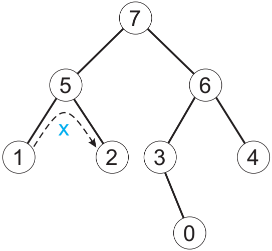
Suchbaum

- Erschöpfende Suche
- Iterativ ebenenweiser Aufbau
- Pro Ebene Entscheidung für oder gegen mindestens ein Element
- Suchbaum maximal k Ebenen tief
  $\rightarrow$ maximal $2 \cdot 2^k$ Knoten

Achtung!

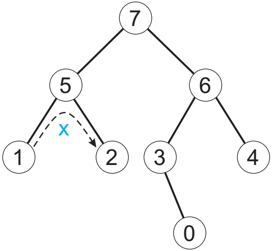
Knoten müssen in polynomieller Zeit konstruier- und prüfbar sein!

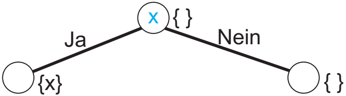
MTO Parameter: Anzahl Paare (2)



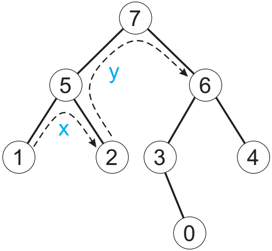
x { }

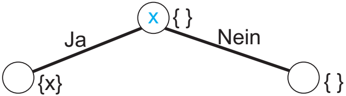
MTO Parameter: Anzahl Paare (2)



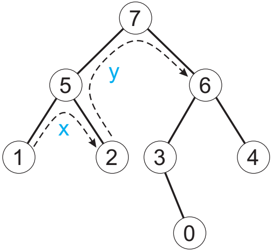


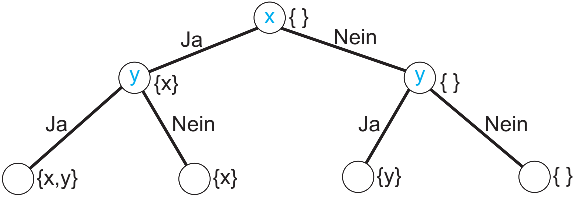
MTO Parameter: Anzahl Paare (2)



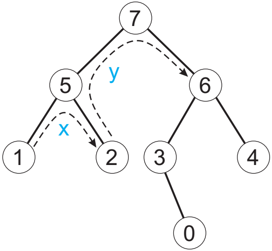


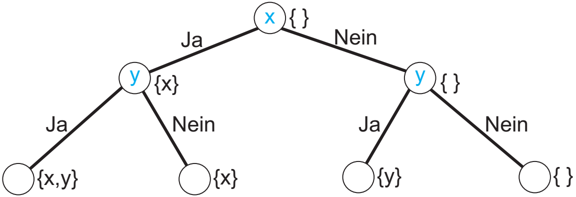
MTO Parameter: Anzahl Paare (2)





MTO Parameter: Anzahl Paare (2)

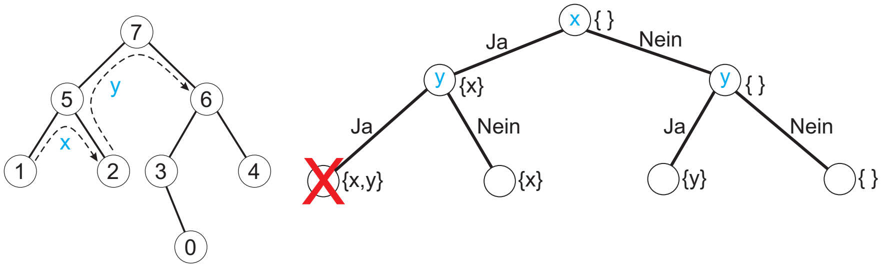




Details

Konsistenz ist mittels einmalig erstellter Hilfskonstruktion in polynomieller Zeit prüfbar.

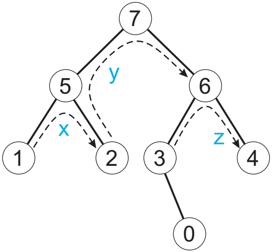
MTO Parameter: Anzahl Paare (2)

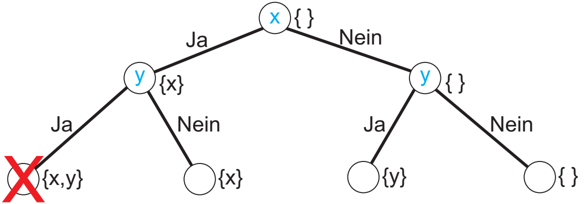


Details

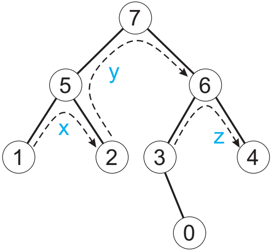
Konsistenz ist mittels einmalig erstellter Hilfskonstruktion in polynomieller Zeit prüfbar.

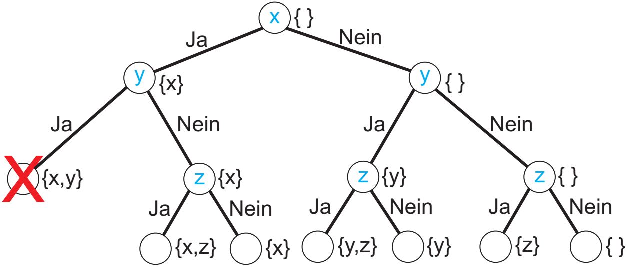
MTO Parameter: Anzahl Paare (2)



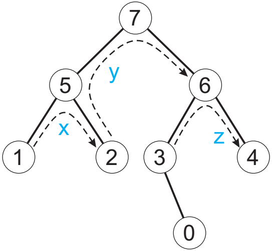


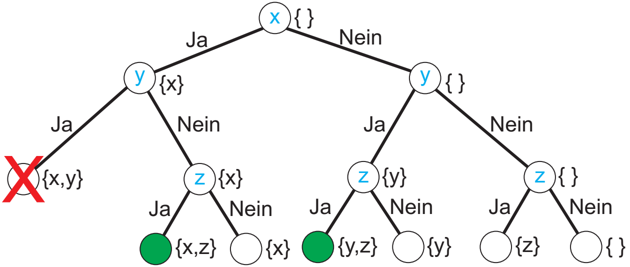
MTO Parameter: Anzahl Paare (2)



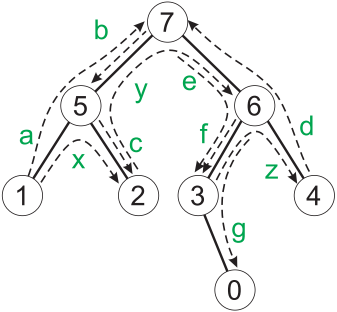


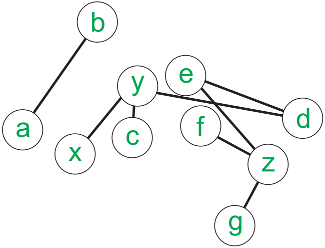
MTO Parameter: Anzahl Paare (2)





Konfliktgraph





Konfliktgraph

- Paare $\rightarrow$ Knoten
- Konflikte $\rightarrow$ Kanten
- Laufzeit: $\mathcal{O}(n^4)$