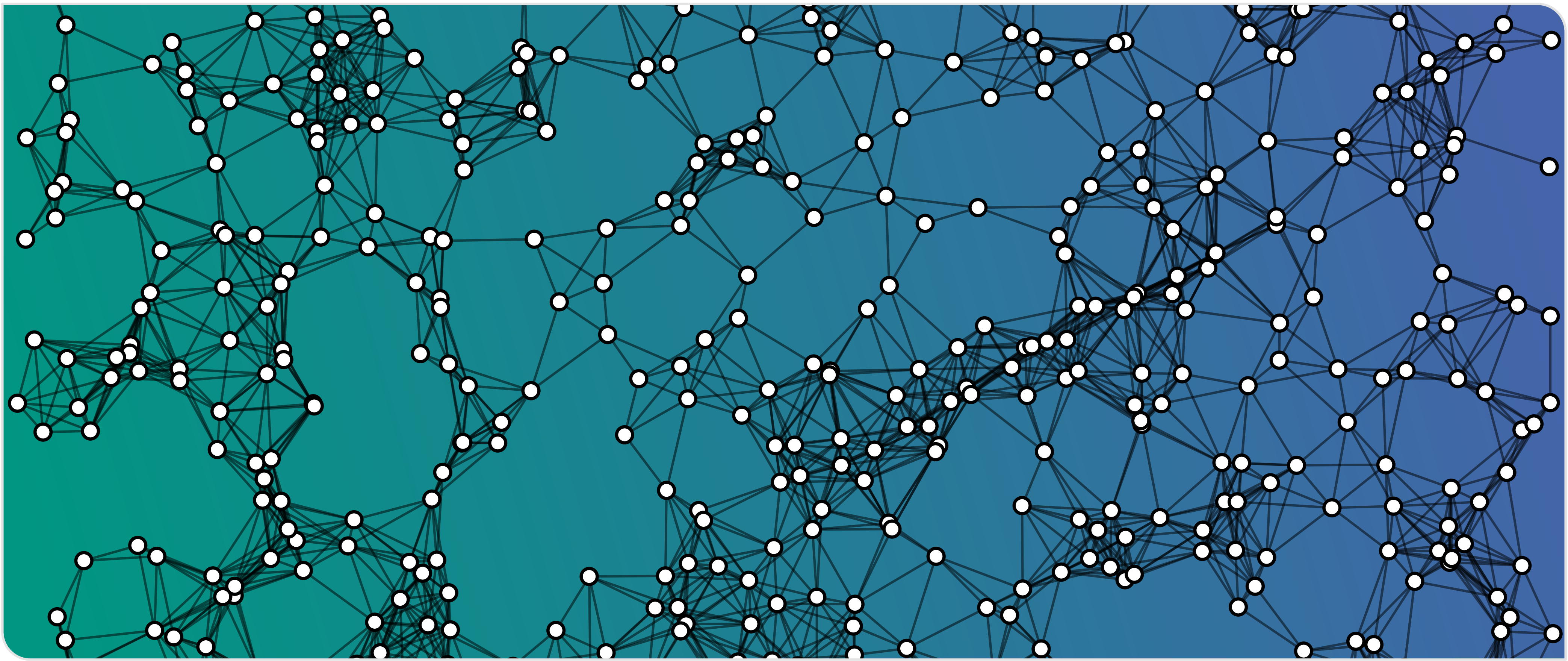
Karlsruher Institut für Technologie

Parametrisierte Algorithmen

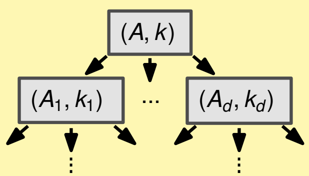
Kernbildung: Ähnliche Bäume

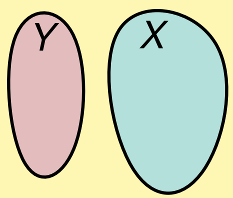


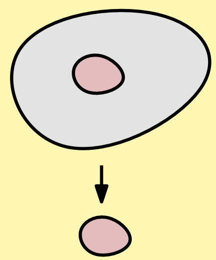
Inhalt

Basic Toolbox

- beschränkte Suchbäume
- iterative Kompression
- Kernbildung

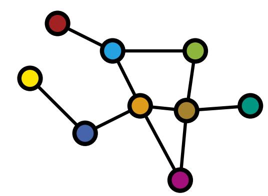


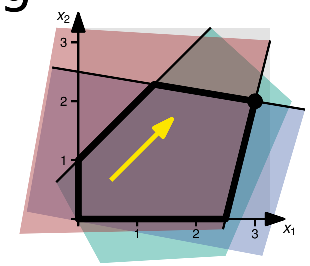


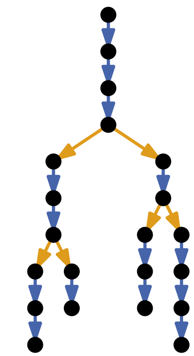


Erweiterte Toolbox

- lineare Programme
- Branch-and-Reduce
- Color Coding





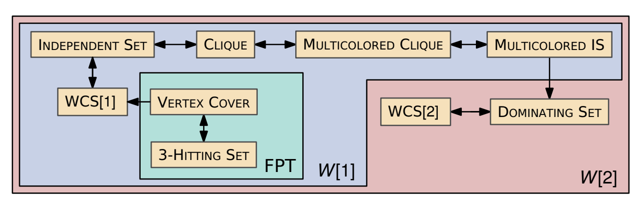


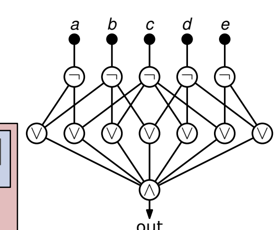
Baumweite

- dynamische Programme
- chordale & planare Graphen
- Courcelles Theorem

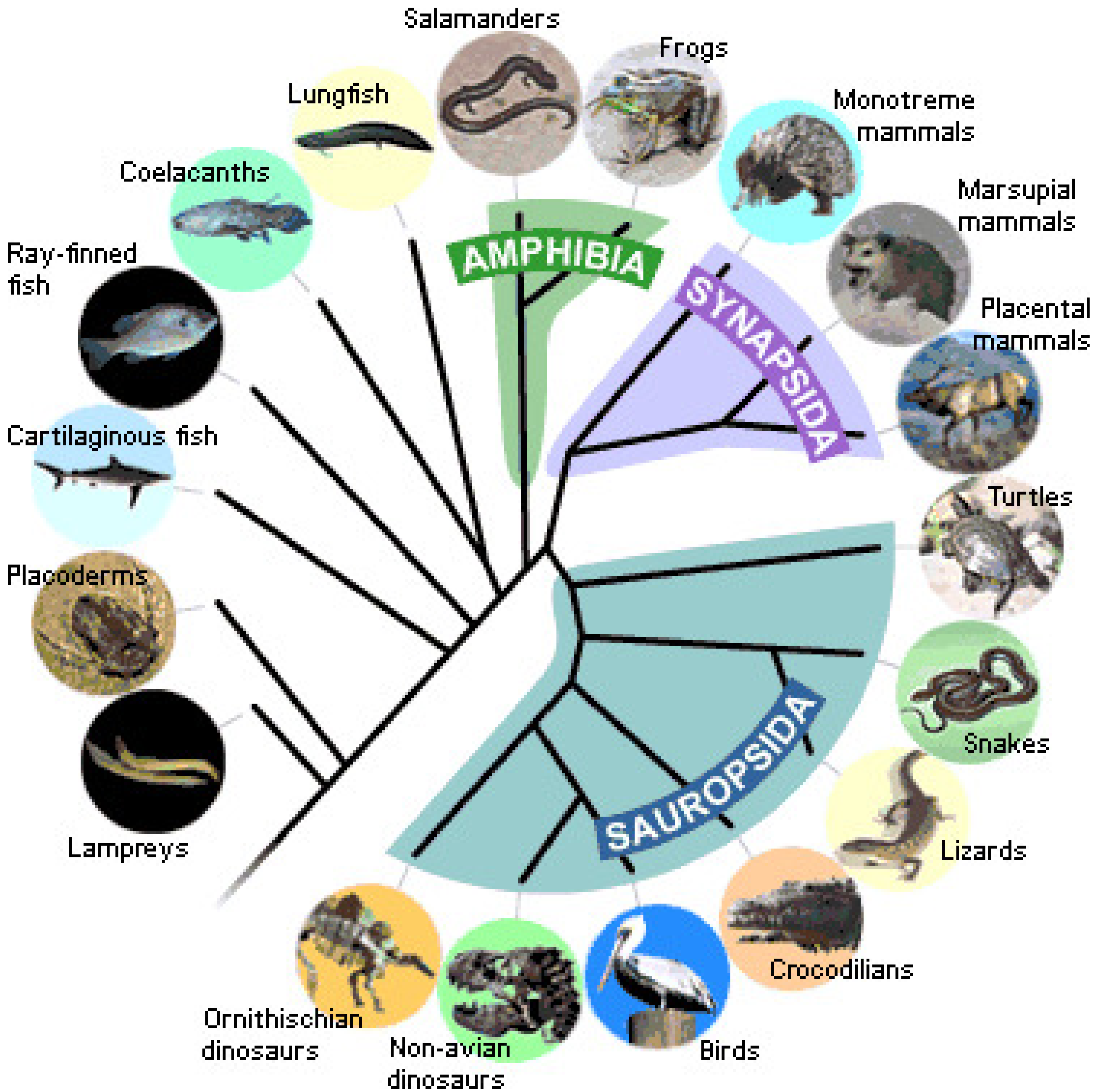
Untere Schranken

- parametrisierte Reduktionen
- boolesche Schaltkreise und die W-Hierarchie
- ETH und SETH

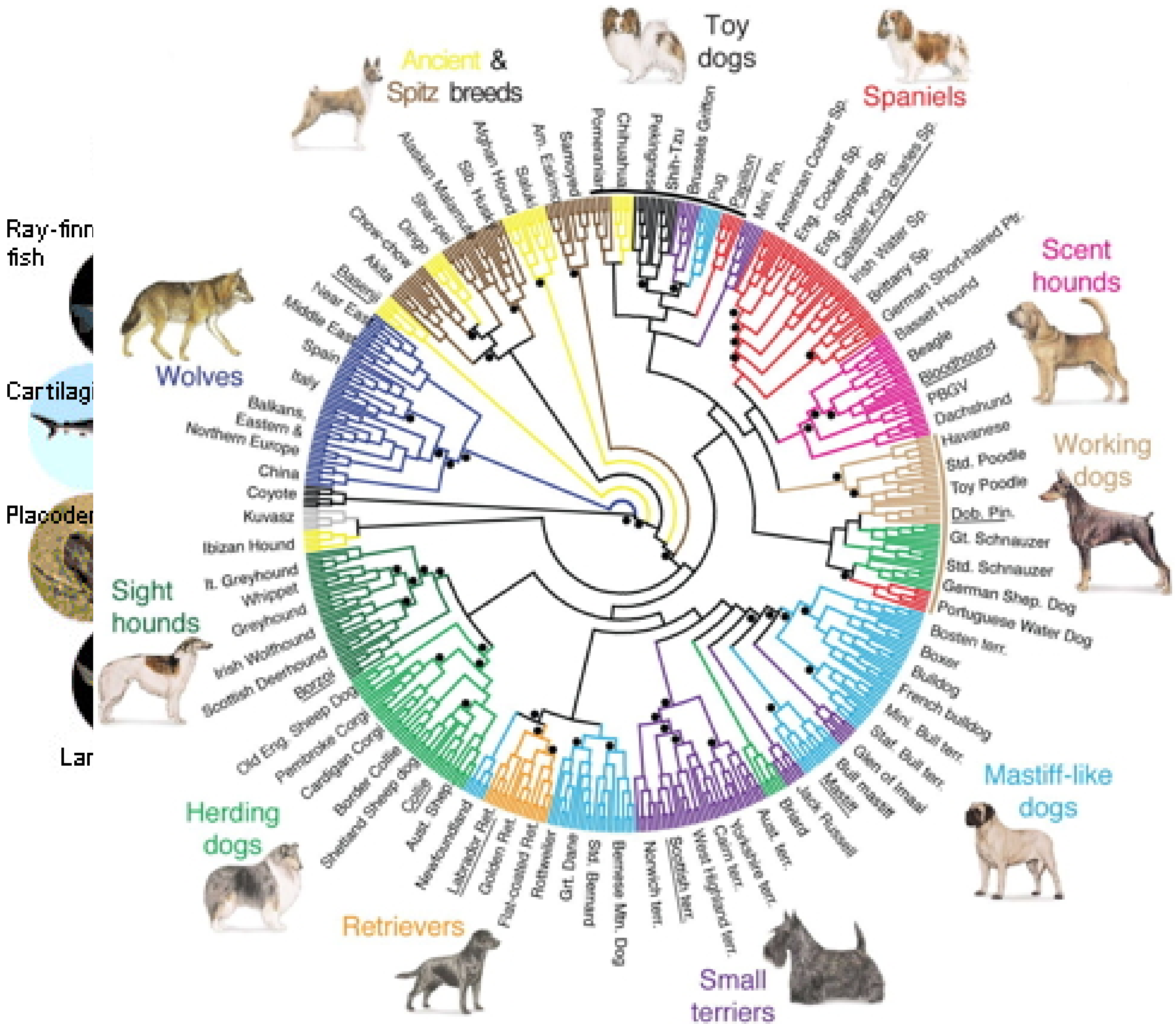




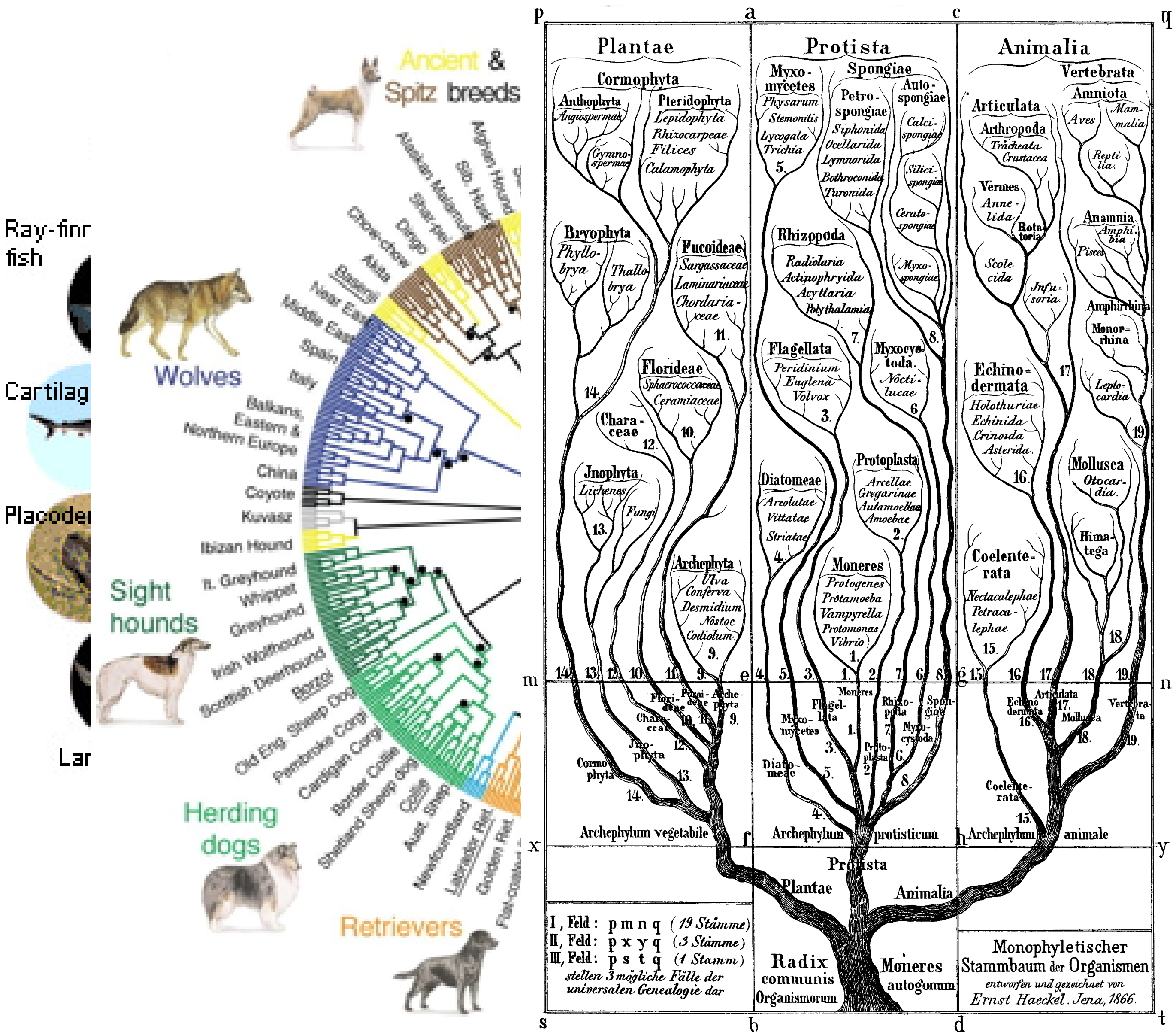
Phylogenetische Bäume



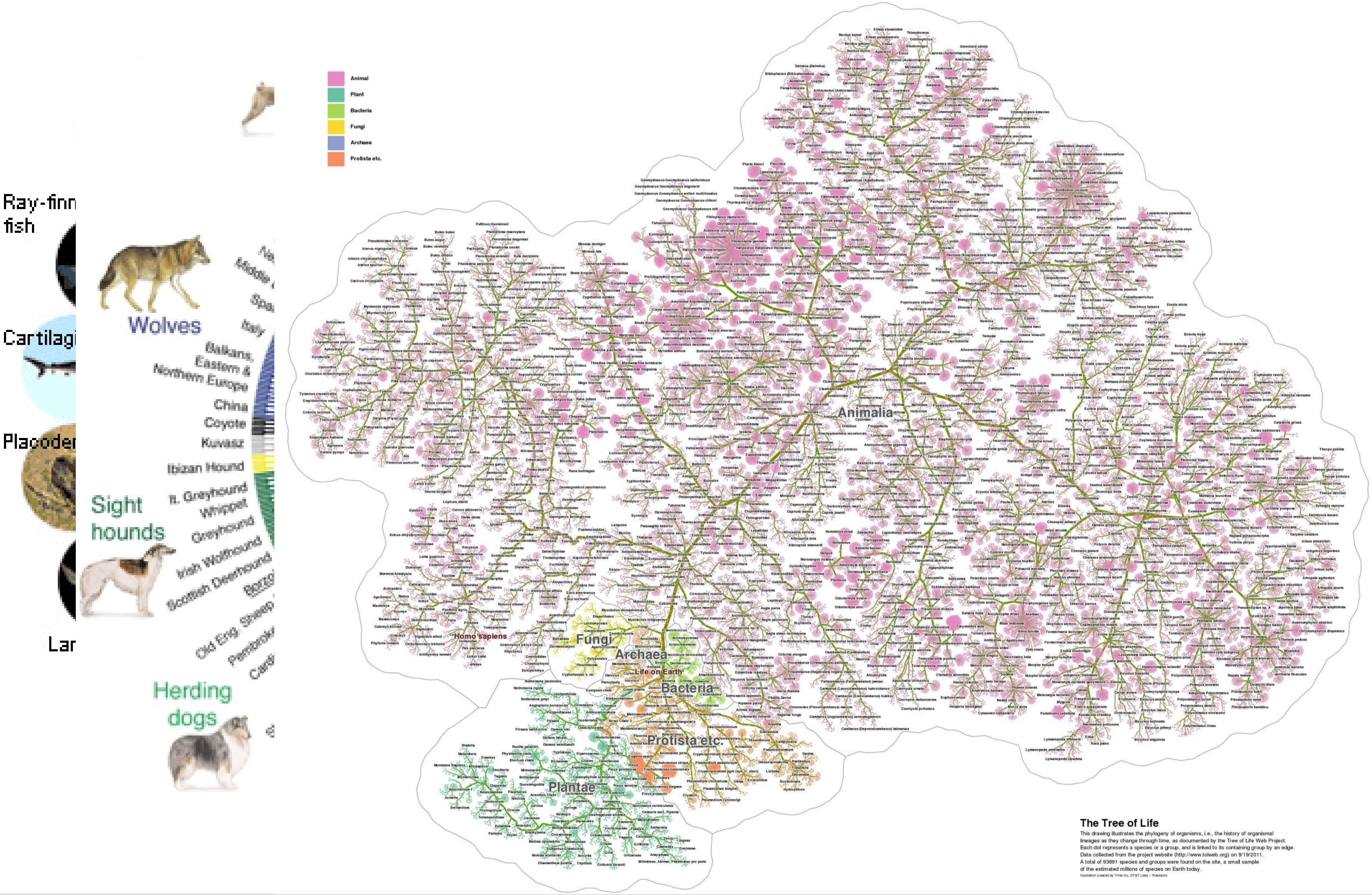
Phylogenetische Bäume



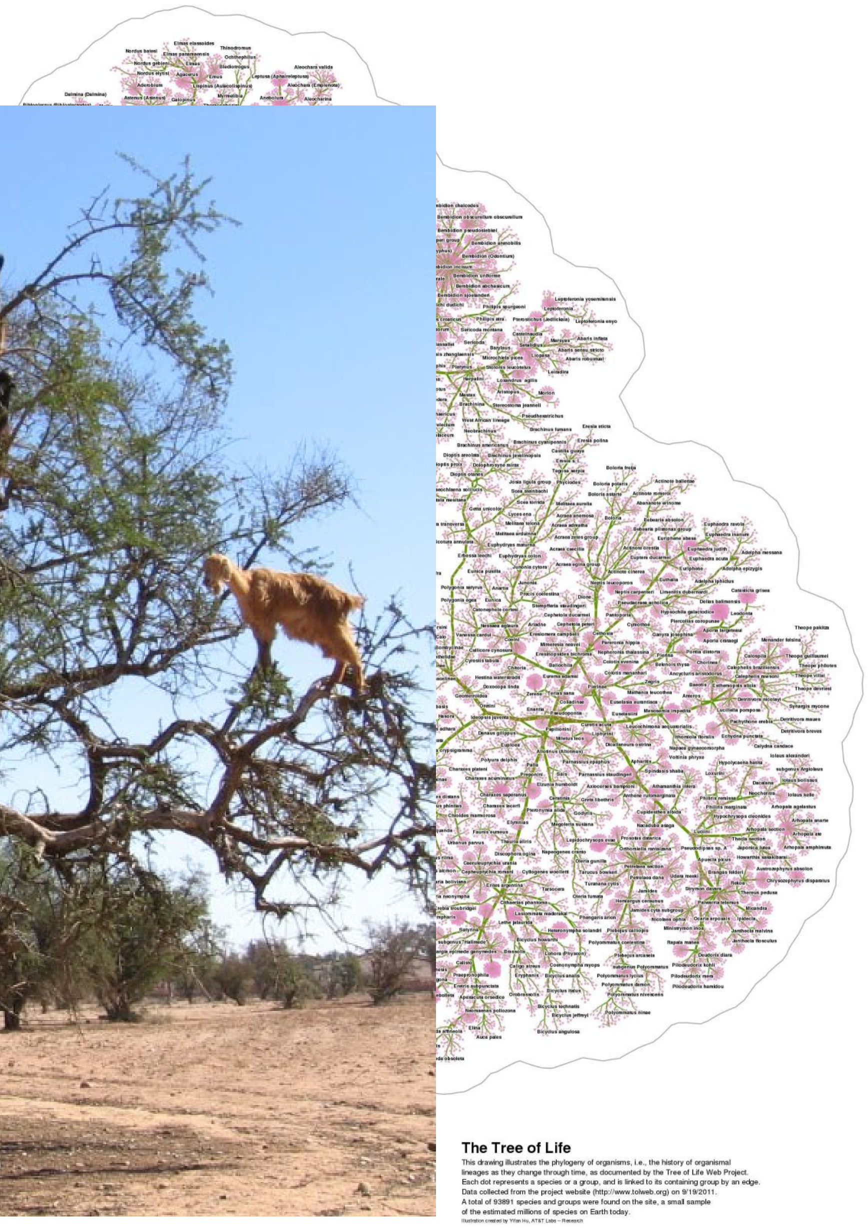
Phylogenetische Bäume



Phylogenetische Bäume



Phylogenetische Bäume



Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

Problem
- unterschiedliche Algorithmen liefern unterschiedliche Bäume

Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

Problem
- unterschiedliche Algorithmen liefern unterschiedliche Bäume
- untersch. Daten (z.B. durch Messfehler) liefern untersch. Bäume

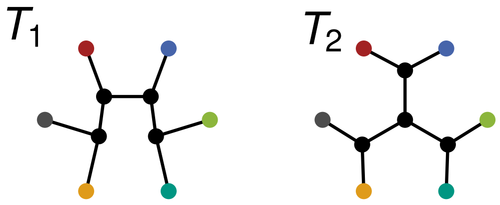
Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

Problem

- unterschiedliche Algorithmen liefern unterschiedliche Bäume
- untersch. Daten (z.B. durch Messfehler) liefern untersch. Bäume
- Wie kann man unterschiedliche Bäume T_1 und T_2 auf der gleichen Blattmenge L miteinander vergleichen?



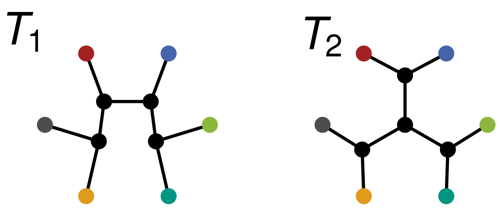
Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

Problem
- unterschiedliche Algorithmen liefern unterschiedliche Bäume
- untersch. Daten (z.B. durch Messfehler) liefern untersch. Bäume
- Wie kann man unterschiedliche Bäume T_1 und T_2 auf der gleichen Blattmenge L miteinander vergleichen?

Maximum Agreement Forest
- Wald F aus Binärbäumen mit Blattmenge L
- Bäume in F enthalten in T_1 und in T_2 (als Minor)



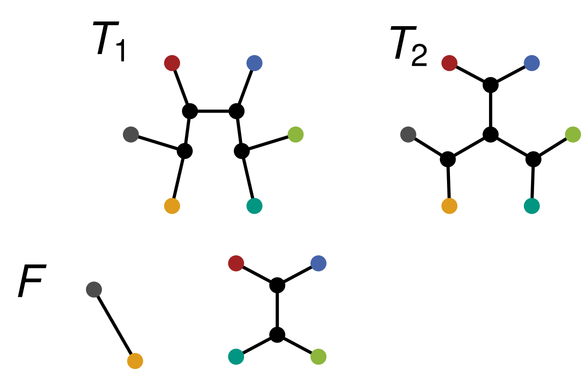
Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

Problem
- unterschiedliche Algorithmen liefern unterschiedliche Bäume
- untersch. Daten (z.B. durch Messfehler) liefern untersch. Bäume
- Wie kann man unterschiedliche Bäume T_1 und T_2 auf der gleichen Blattmenge L miteinander vergleichen?

Maximum Agreement Forest
- Wald F aus Binärbäumen mit Blattmenge L
- Bäume in F enthalten in T_1 und in T_2 (als Minor)



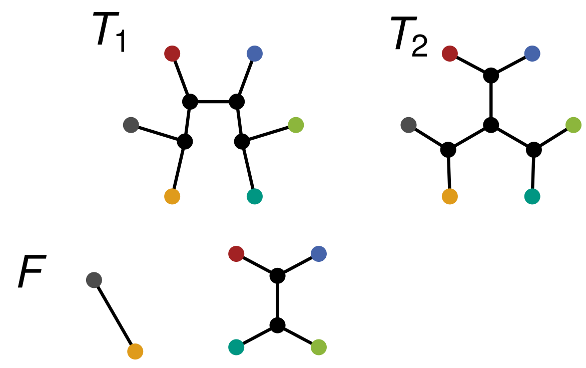
Phylogenetische Bäume

Definition
Ein **phylogenetischer Baum** ist ein ungewurzelter und vollständiger Binärbaum, sodass jedes Blatt ein eindeutiges Label ($\equiv$ Spezies) hat.

- werden oft automatisiert erstellt
- beispielsweise basierend auf DNA-Sequenzierung

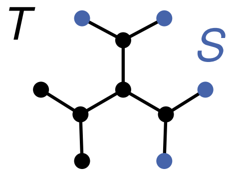
Problem
- unterschiedliche Algorithmen liefern unterschiedliche Bäume
- untersch. Daten (z.B. durch Messfehler) liefern untersch. Bäume
- Wie kann man unterschiedliche Bäume T_1 und T_2 auf der gleichen Blattmenge L miteinander vergleichen?

Maximum Agreement Forest
- Wald F aus Binärbäumen mit Blattmenge L
- Bäume in F enthalten in T_1 und in T_2 (als Minor)
- minimiere #Bäume in F



Notation

Blattinduzierte Teilbäume

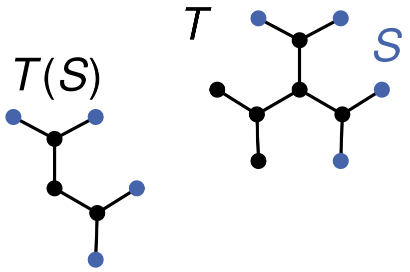


- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$

Notation

Blattinduzierte Teilbäume

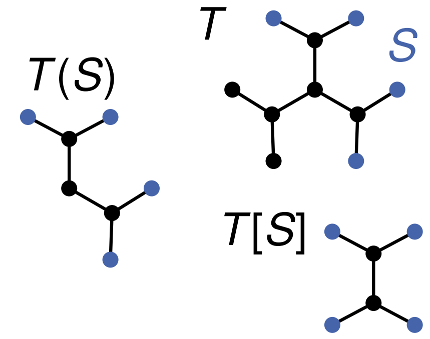
- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält



Notation

Blattinduzierte Teilbäume

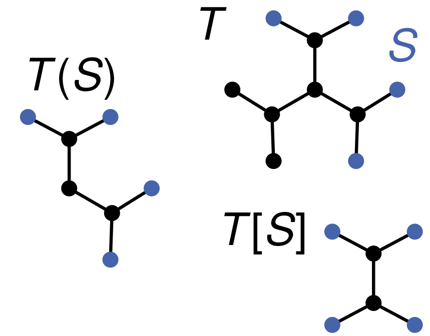
- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält
- kontrahiere alle Knoten mit Grad 2 in $T(S) \rightarrow T[S]$



Notation

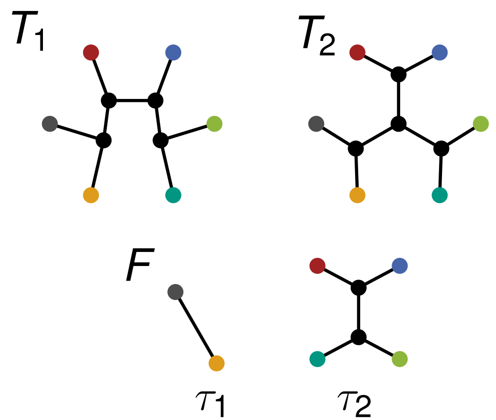
Blattinduzierte Teilbäume

- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält
- kontrahiere alle Knoten mit Grad 2 in $T(S) \to T[S]$



Agreement Forest

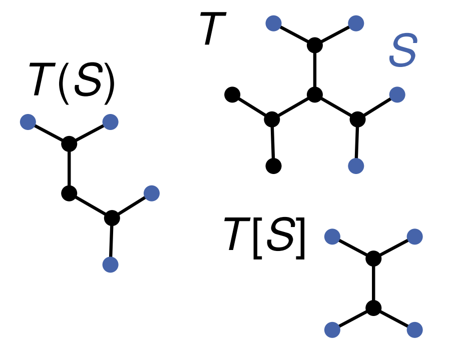
- sei F ein Wald bestehend aus den Bäumen $\tau_1, \dots, \tau_k$
- betrachte zwei Bäume T_1 und T_2 mit $L(T_1) = L(T_2) = L(F)$
- F ist *Agreement Forest* für T_1 und T_2, wenn $T_1[L(\tau_i)] = T_2[L(\tau_i)] = \tau_i$ und die $T_1(L(\tau_i))$ (für $i \in \{1, \dots, k\}$), sowie die $T_2(L(\tau_i))$ sind disjunkt



Notation

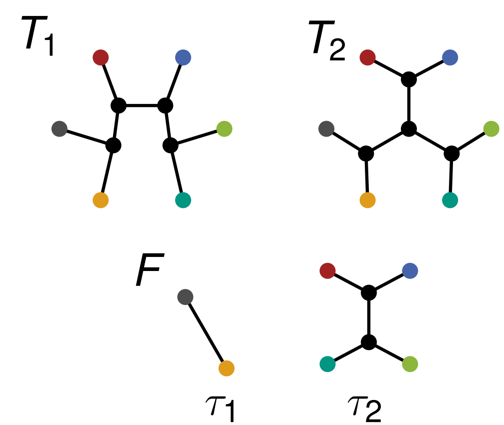
Blattinduzierte Teilbäume

- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält
- kontrahiere alle Knoten mit Grad 2 in $T(S) \to T[S]$

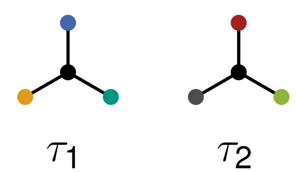


Agreement Forest

- sei F ein Wald bestehend aus den Bäumen $\tau_1, \dots, \tau_k$
- betrachte zwei Bäume T_1 und T_2 mit $L(T_1) = L(T_2) = L(F)$
- F ist *Agreement Forest* für T_1 und T_2, wenn $T_1[L(\tau_i)] = T_2[L(\tau_i)] = \tau_i$ und die $T_1(L(\tau_i))$ (für $i \in \{1, \dots, k\}$), sowie die $T_2(L(\tau_i))$ sind disjunkt



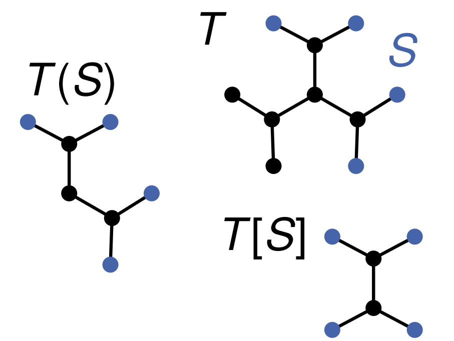
Disjunktheit verletzt:



Notation

Blattinduzierte Teilbäume

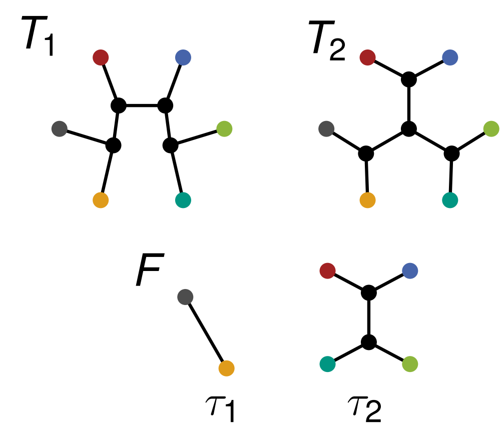
- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält
- kontrahiere alle Knoten mit Grad 2 in $T(S) \to T[S]$



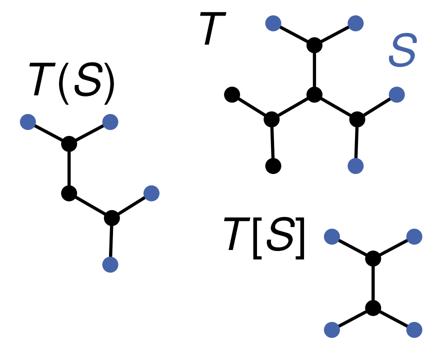
Agreement Forest

- sei F ein Wald bestehend aus den Bäumen $\tau_1, \dots, \tau_k$
- betrachte zwei Bäume T_1 und T_2 mit $L(T_1) = L(T_2) = L(F)$
- F ist *Agreement Forest* für T_1 und T_2, wenn $T_1[L(\tau_i)] = T_2[L(\tau_i)] = \tau_i$ und die $T_1(L(\tau_i))$ (für $i \in \{1, \dots, k\}$), sowie die $T_2(L(\tau_i))$ sind disjunkt

Problem: MAXIMUM AGREEMENT FOREST
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Notation



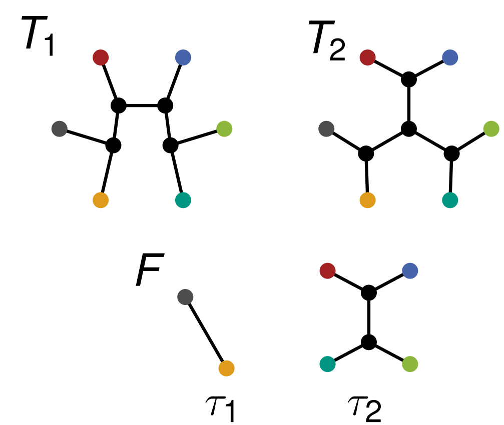
Blattinduzierte Teilbäume

- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält
- kontrahiere alle Knoten mit Grad 2 in $T(S) \rightarrow T[S]$

Agreement Forest

- sei F ein Wald bestehend aus den Bäumen $\tau_1, \dots, \tau_k$
- betrachte zwei Bäume T_1 und T_2 mit $L(T_1) = L(T_2) = L(F)$
- F ist *Agreement Forest* für T_1 und T_2, wenn $T_1[L(\tau_i)] = T_2[L(\tau_i)] = \tau_i$ und die $T_1(L(\tau_i))$ (für $i \in \{1, \dots, k\}$), sowie die $T_2(L(\tau_i))$ sind disjunkt

> **Problem: Maximum Agreement Forest**
> Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



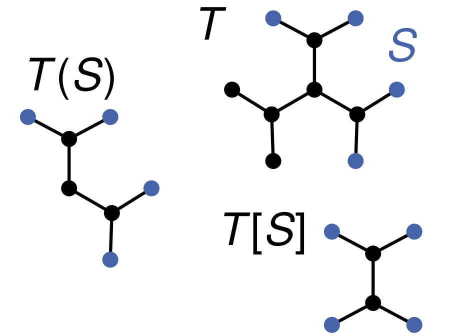
Ziel

- finde einen Kernbildungsalgorithmus

Notation

Blattinduzierte Teilbäume

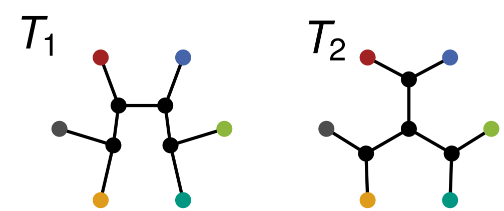
- sei T ein Baum mit Blättern $L(T)$ und sei $S \subseteq L(T)$
- $T(S)$ = minimaler Teilbaum von T, der S enthält
- kontrahiere alle Knoten mit Grad 2 in $T(S) \to T[S]$



Agreement Forest

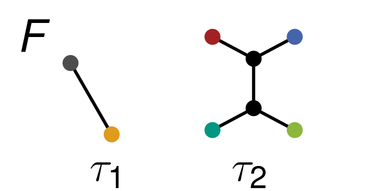
- sei F ein Wald bestehend aus den Bäumen $\tau_1, \dots, \tau_k$
- betrachte zwei Bäume T_1 und T_2 mit $L(T_1) = L(T_2) = L(F)$
- F ist *Agreement Forest* für T_1 und T_2, wenn $T_1[L(\tau_i)] = T_2[L(\tau_i)] = \tau_i$ und die $T_1(L(\tau_i))$ (für $i \in \{1, \dots, k\}$), sowie die $T_2(L(\tau_i))$ sind disjunkt

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



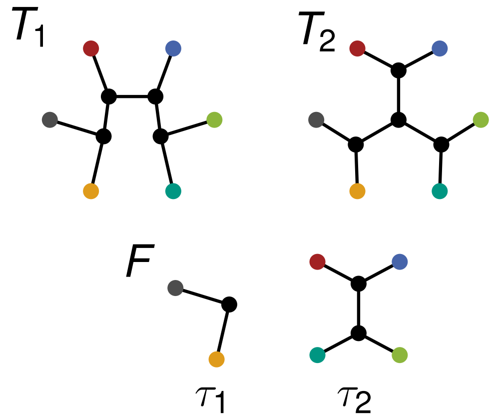
Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab



Grober Fahrplan

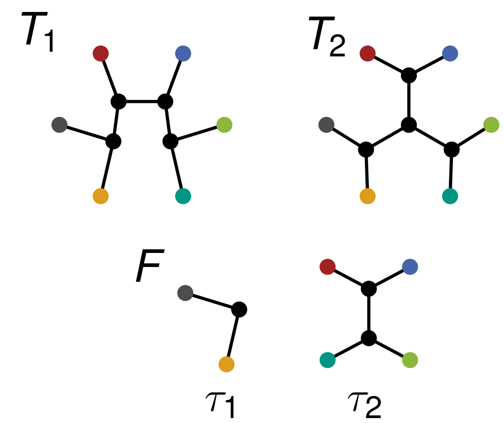
Problem: MAXIMUM AGREEMENT FOREST
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel
- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

Grober Fahrplan

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



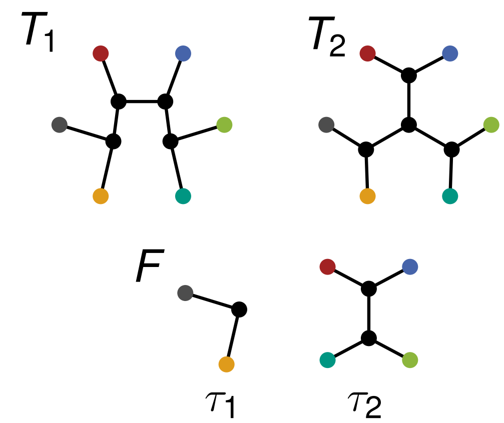
Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

Vorschläge?

Grober Fahrplan

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel

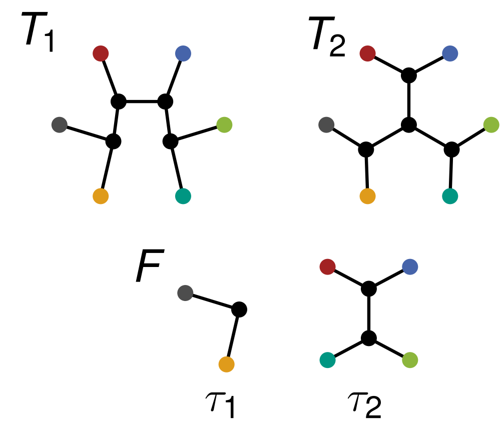
- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

Vorschläge?

Bevor wir losreduzieren: Was ist das Ziel?

Grober Fahrplan

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

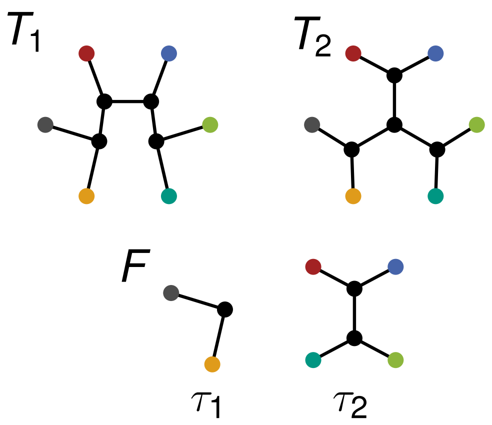
Vorschläge?

Bevor wir losreduzieren: Was ist das Ziel?

- Größe der resultierenden Bäume hängt nur von k ab

Grober Fahrplan

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

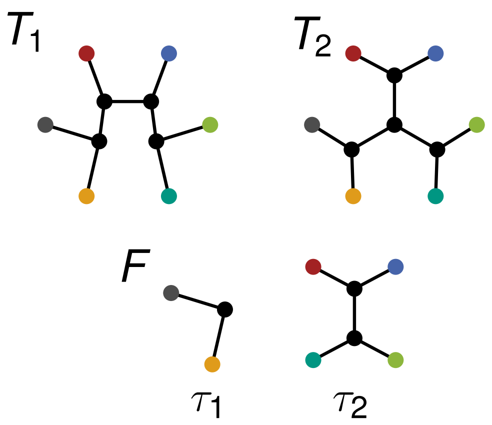
Vorschläge?

Bevor wir losreduzieren: Was ist das Ziel?

- Größe der resultierenden Bäume hängt nur von k ab
- etwas stärkere, aber konkretere Behauptung:

Grober Fahrplan

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

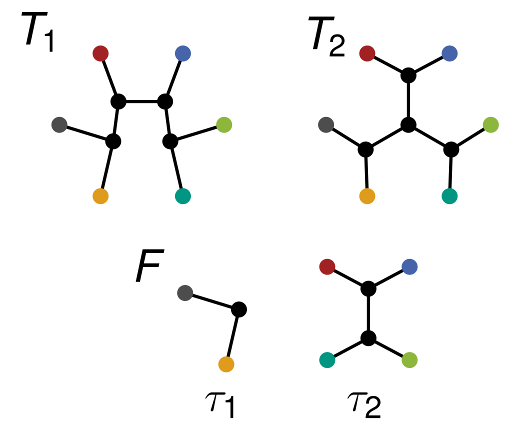
Vorschläge?

Bevor wir losreduzieren: Was ist das Ziel?

- Größe der resultierenden Bäume hängt nur von k ab
- etwas stärkere, aber konkretere Behauptung:
  - jeder Baum in einem Agreement Forest enthält nur wenige Blätter

Grober Fahrplan

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

Vorschläge?

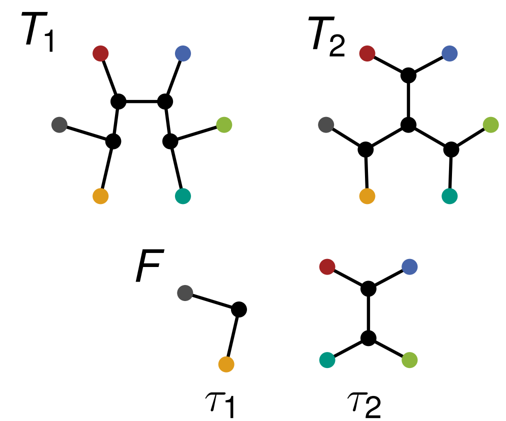
Bevor wir losreduzieren: Was ist das Ziel?

- Größe der resultierenden Bäume hängt nur von k ab
- etwas stärkere, aber konkretere Behauptung:
  - jeder Baum in einem Agreement Forest enthält nur wenige Blätter

  $\Rightarrow$ insgesamt wenige Blätter $\Rightarrow$ kleine Instanz

Grober Fahrplan

Problem: MAXIMUM AGREEMENT FOREST
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?



Ziel

- finde einen Kernbildungsalgorithmus
- also polynomielle Reduktionsregeln
- Größe der resultierenden Bäume hängt nur von k ab

Vorschläge?

Bevor wir losreduzieren: Was ist das Ziel?

- Größe der resultierenden Bäume hängt nur von k ab
- etwas stärkere, aber konkretere Behauptung:
  - jeder Baum in einem Agreement Forest enthält nur wenige Blätter
    $\Rightarrow$ insgesamt wenige Blätter $\Rightarrow$ kleine Instanz
- Welche Baumstrukturen widersprechen der Behauptung?
- Können wir diese mittels Reduktionsregeln loswerden?

Ungünstige Baumstruktur 1

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

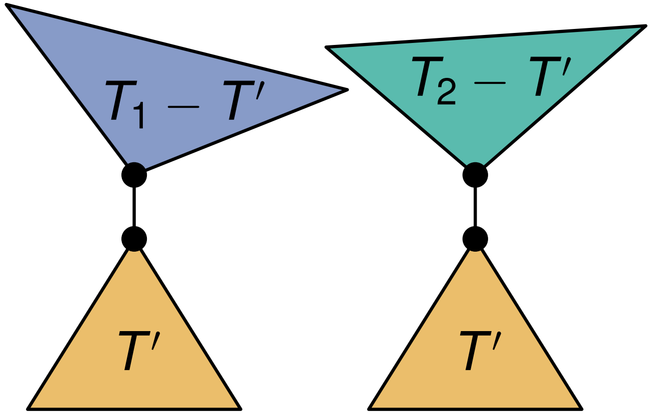
Gegenbeispiel

Ungünstige Baumstruktur 1

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

- T_1 und T_2 können einen Teilbaum T' mit vielen Blättern gemeinsam haben
- im Agreement Forest kann man T' als einen der Bäume $\tau_1, \dots, \tau_k$ wählen

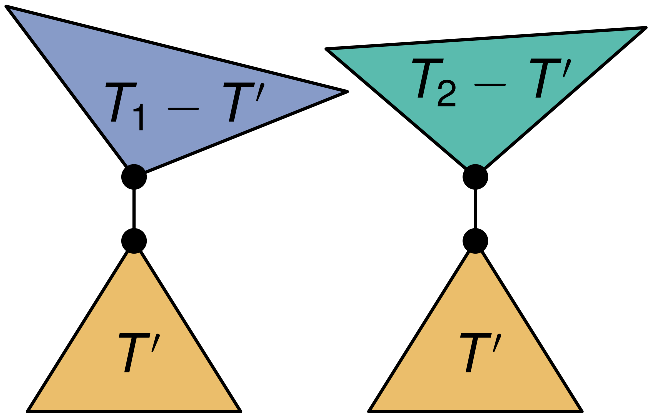


Ungünstige Baumstruktur 1

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

- T_1 und T_2 können einen Teilbaum T' mit vielen Blättern gemeinsam haben
- im Agreement Forest kann man T' als einen der Bäume $\tau_1, \dots, \tau_k$ wählen

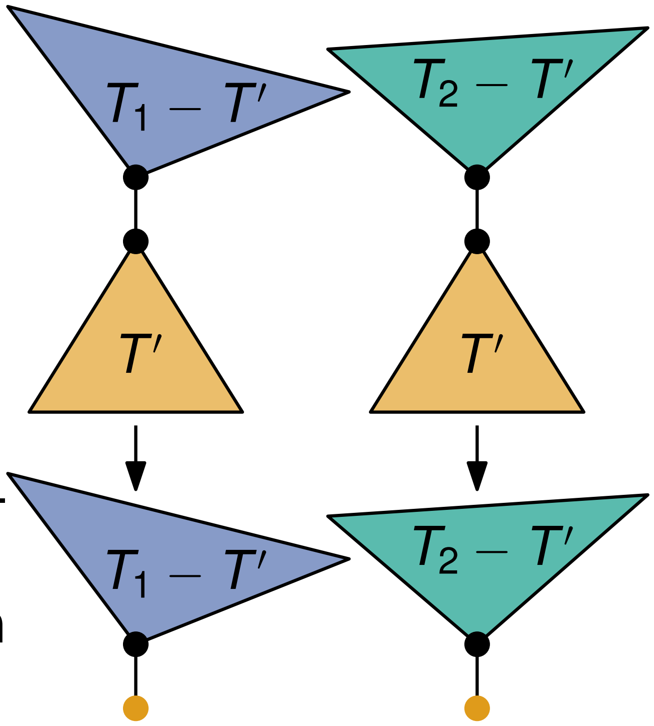


Reduktionsregel 1

Ungünstige Baumstruktur 1

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel
- T_1 und T_2 können einen Teilbaum T' mit vielen Blättern gemeinsam haben
- im Agreement Forest kann man T' als einen der Bäume $\tau_1, \ldots, \tau_k$ wählen



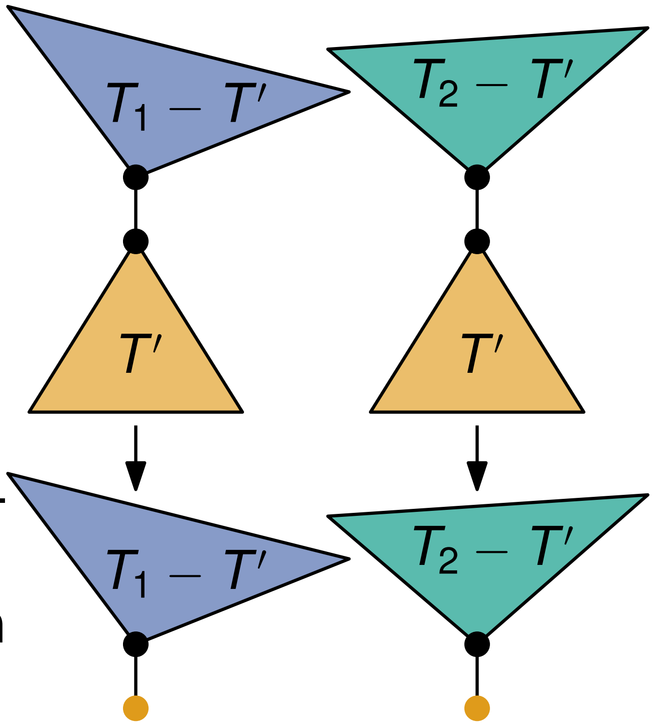
Reduktionsregel 1
- finde Kanten in T_1 und T_2, sodass diese den selben Baum T' abtrennen
- ersetze T' durch ein einzelnes Blatt mit neuem (gleichem) Label

Ungünstige Baumstruktur 1

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

- T_1 und T_2 können einen Teilbaum T' mit vielen Blättern gemeinsam haben
- im Agreement Forest kann man T' als einen der Bäume $\tau_1, \ldots, \tau_k$ wählen



Reduktionsregel 1

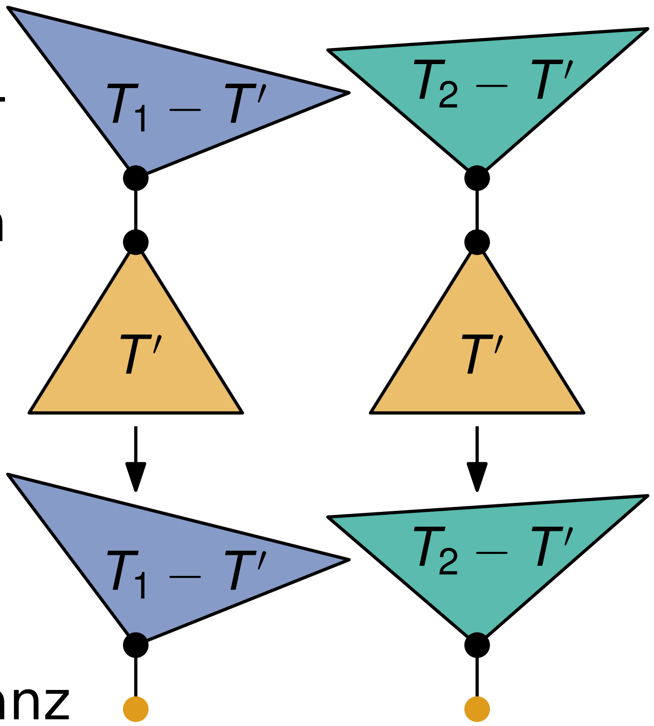
- finde Kanten in T_1 und T_2, sodass diese den selben Baum T' abtrennen
- ersetze T' durch ein einzelnes Blatt mit neuem (gleichem) Label

Lemma
Reduktionsregel 1 ist sicher und in polynomieller Zeit ausführbar.

Beweis: nächste Folie

Reduktionsregel 1

- finde Kanten in T_1 und T_2, sodass diese den selben Baum T' abtrennen
- ersetze T' durch ein einzelnes Blatt mit neuem (gleichem) Label



Lemma
Reduktionsregel 1 ist sicher und in polynomieller Zeit ausführbar.

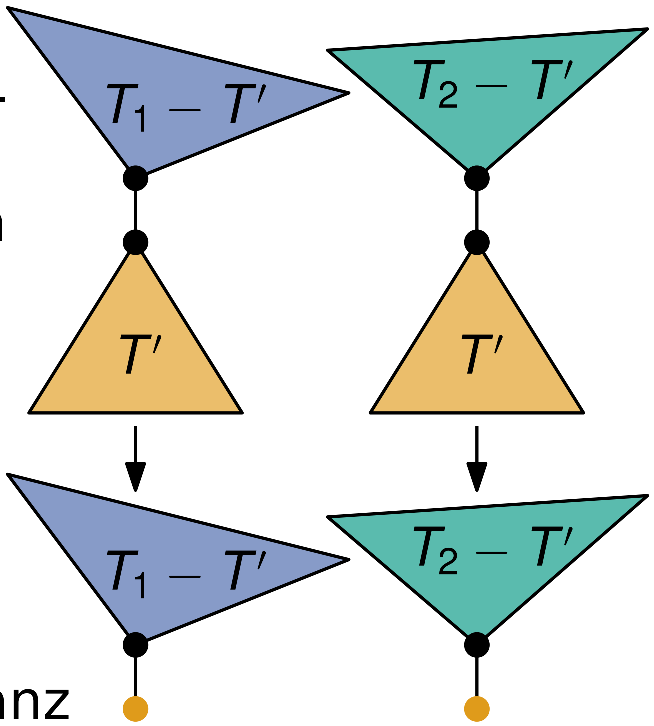
Beweis

- sei F' ein Agreement Forest der neuen Instanz und sei τ_1' der Baum mit dem neuen Blatt in F'

Reductionsregel 1

Reduktionsregel 1

- finde Kanten in T_1 und T_2, sodass diese den selben Baum T' abtrennen
- ersetze T' durch ein einzelnes Blatt mit neuem (gleichem) Label



Lemma
Reduktionsregel 1 ist sicher und in polynomieller Zeit ausführbar.

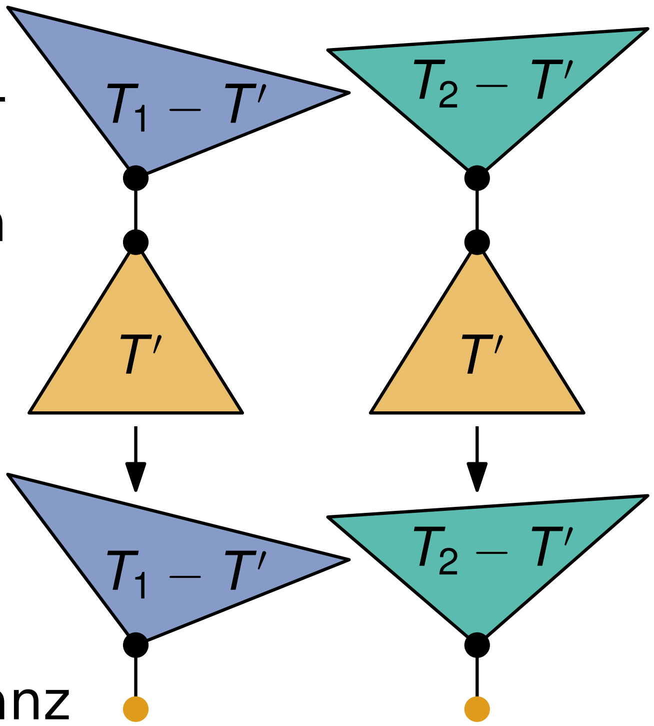
Beweis

- sei F' ein Agreement Forest der neuen Instanz und sei τ_1' der Baum mit dem neuen Blatt in F'
- Ersetzung dieses Blattes in τ_1' durch T' liefert Agreement Forest der gleichen Größe für T_1 und T_2

Reductionsregel 1

Reduktionsregel 1

- finde Kanten in T_1 und T_2, sodass diese den selben Baum T' abtrennen
- ersetze T' durch ein einzelnes Blatt mit neuem (gleichem) Label



Lemma
Reduktionsregel 1 ist sicher und in polynomieller Zeit ausführbar.

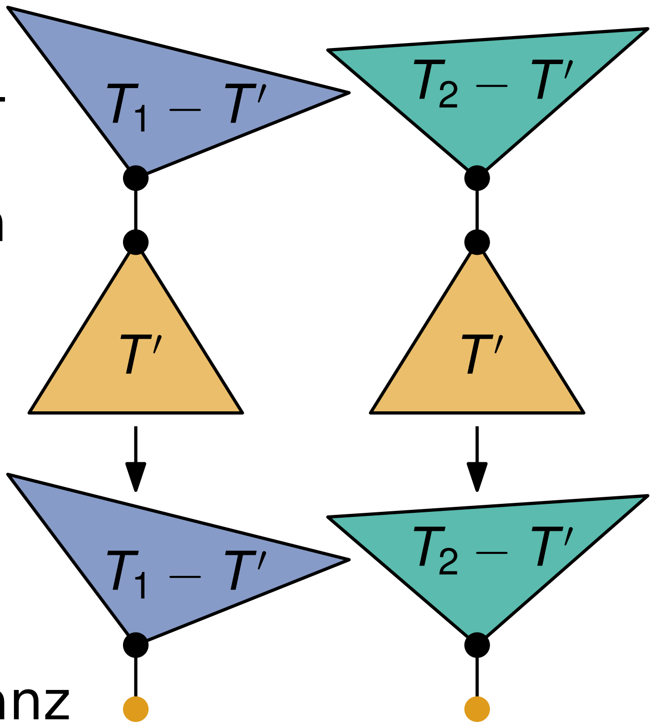
Beweis

- sei F' ein Agreement Forest der neuen Instanz und sei τ_1' der Baum mit dem neuen Blatt in F'
- Ersetzung dieses Blattes in τ_1' durch T' liefert Agreement Forest der gleichen Größe für T_1 und T_2
- andere Richtung: ähnlich

Reductionsregel 1

Reduktionsregel 1

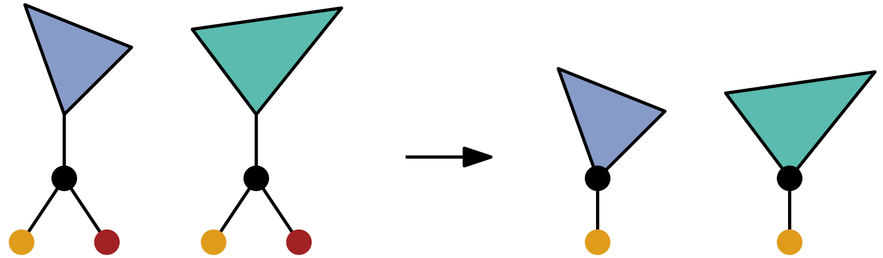
- finde Kanten in T_1 und T_2, sodass diese den selben Baum T' abtrennen
- ersetze T' durch ein einzelnes Blatt mit neuem (gleichem) Label



Lemma
Reduktionsregel 1 ist sicher und in polynomieller Zeit ausführbar.

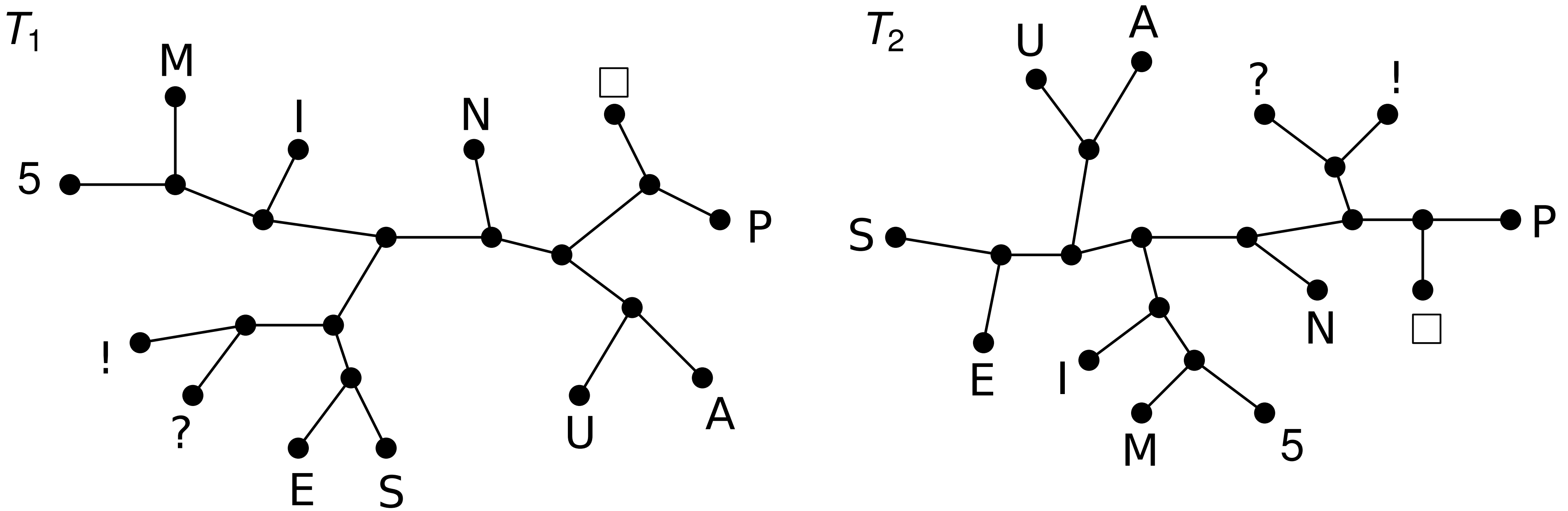
Beweis

- sei F' ein Agreement Forest der neuen Instanz und sei τ_1' der Baum mit dem neuen Blatt in F'
- Ersetzung dieses Blattes in τ_1' durch T' liefert Agreement Forest der gleichen Größe für T_1 und T_2
- andere Richtung: ähnlich
- Laufzeit: z.B. durch iterative Ersetzung von „Kirschen“



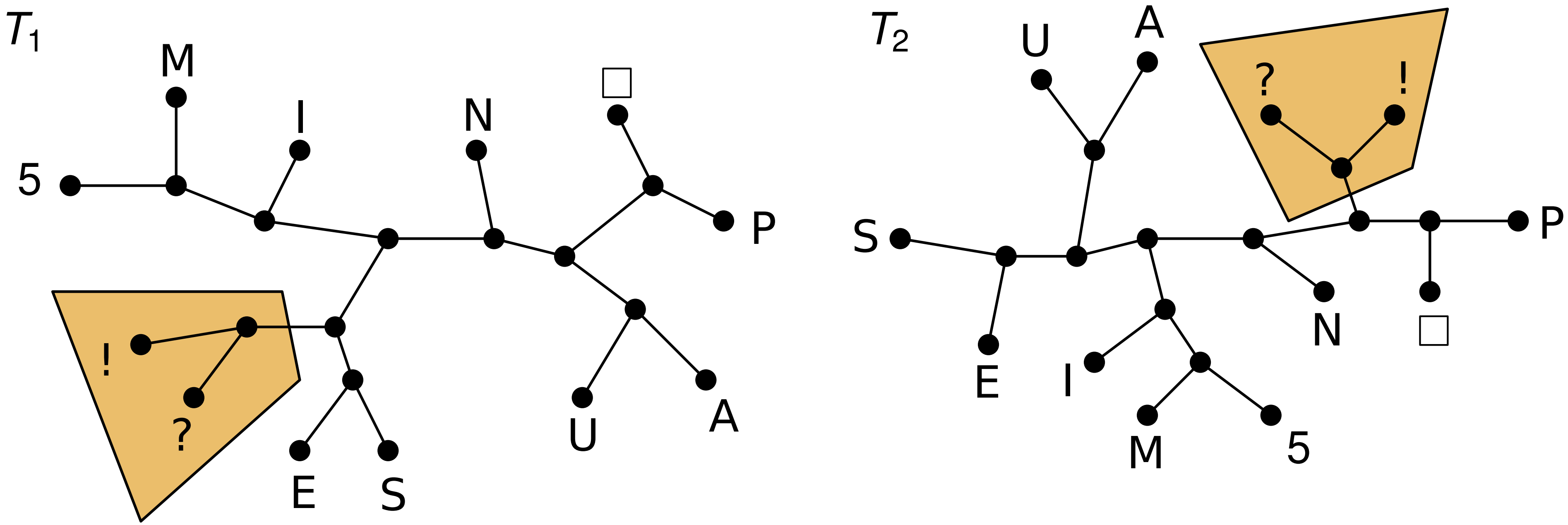
Beispiel

Löse diese Maximum Agreement Forest-Instanz



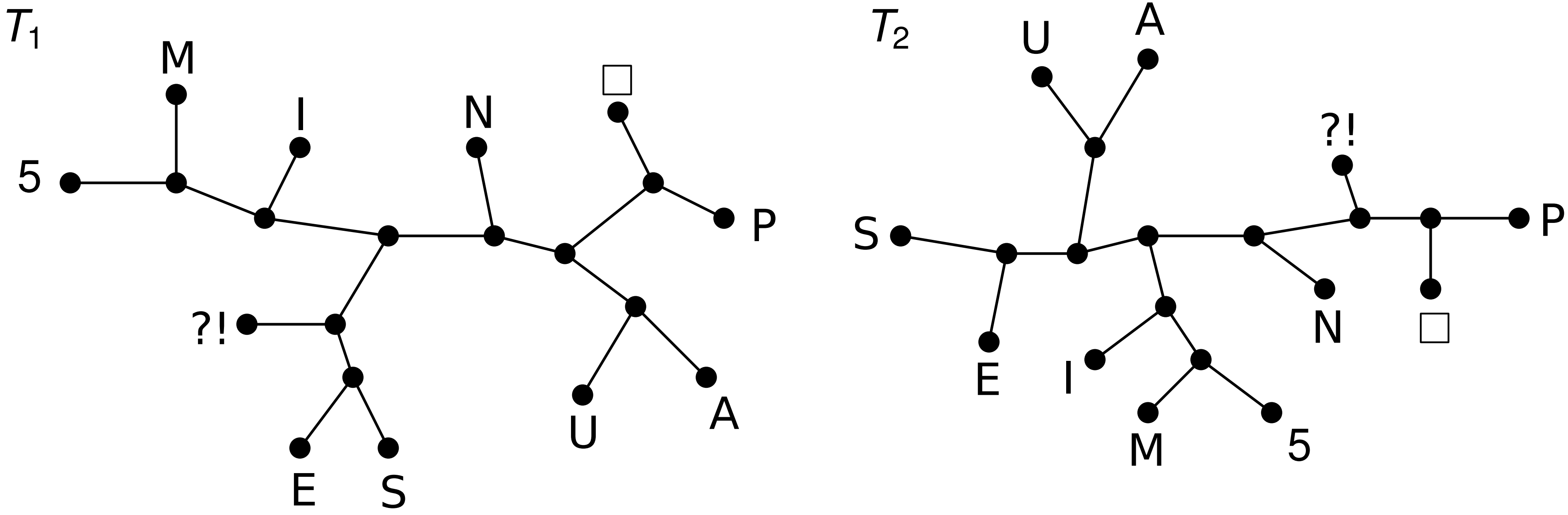
Beispiel

Löse diese Maximum Agreement Forest-Instanz



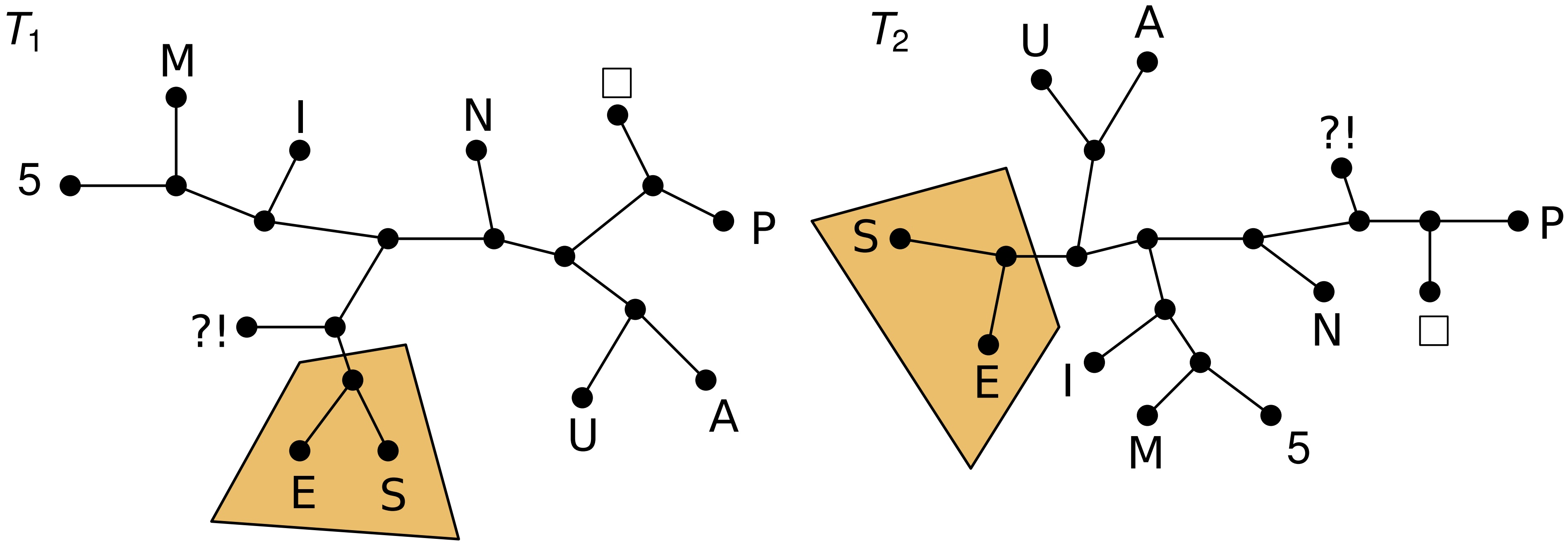
Beispiel

Löse diese Maximum Agreement Forest-Instanz



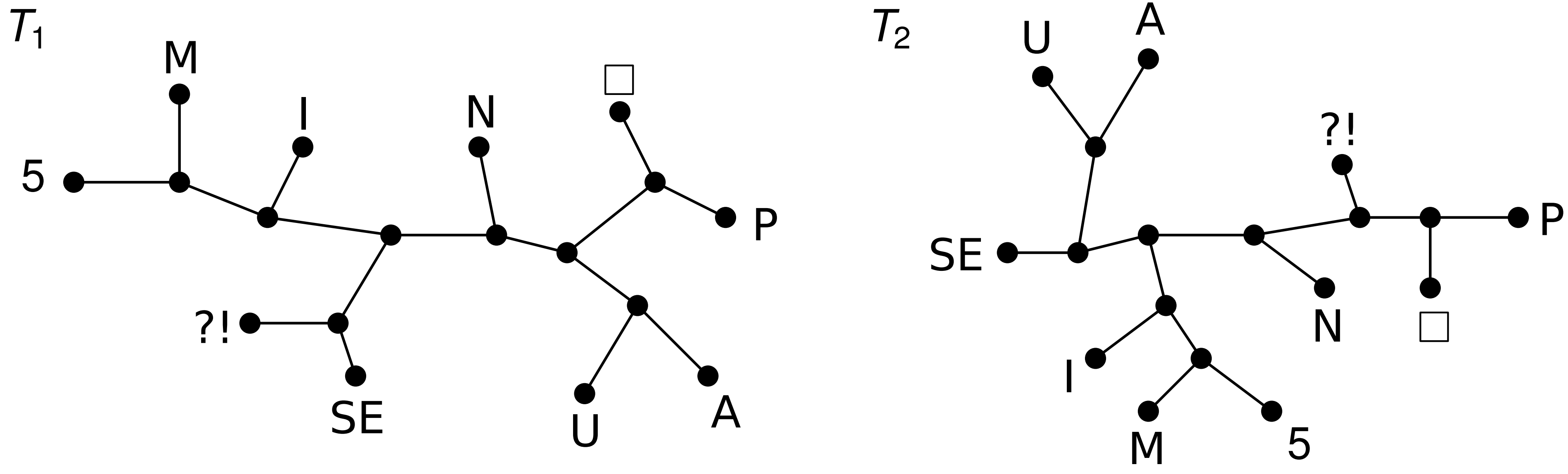
Beispiel

Löse diese Maximum Agreement Forest-Instanz



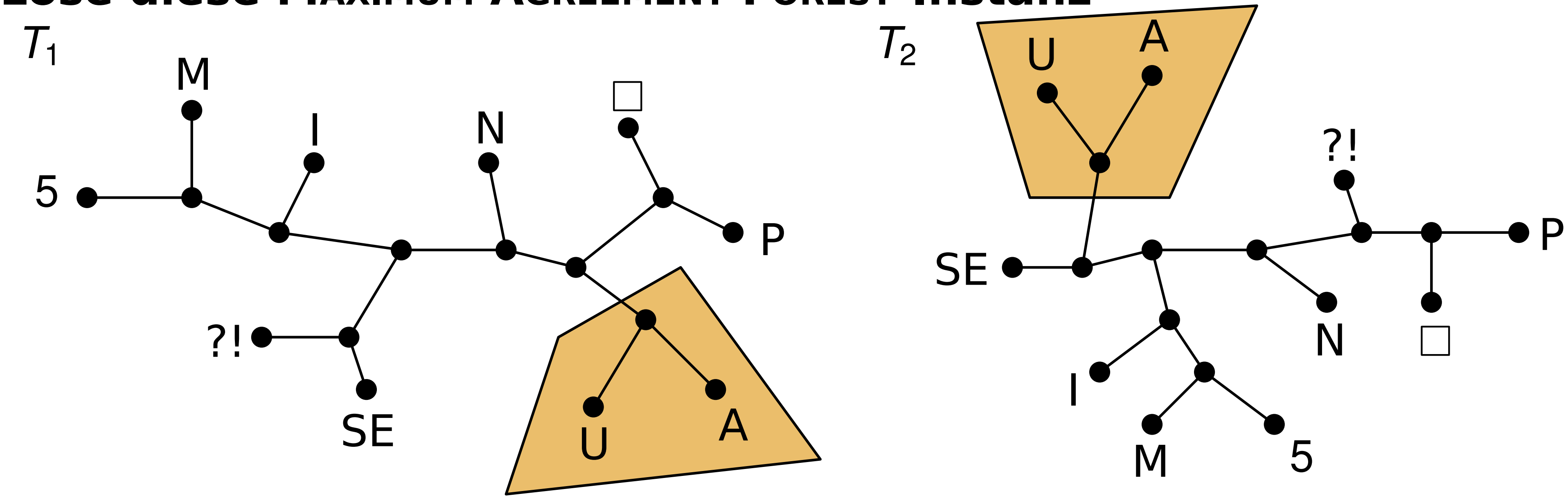
Beispiel

Löse diese Maximum Agreement Forest-Instanz



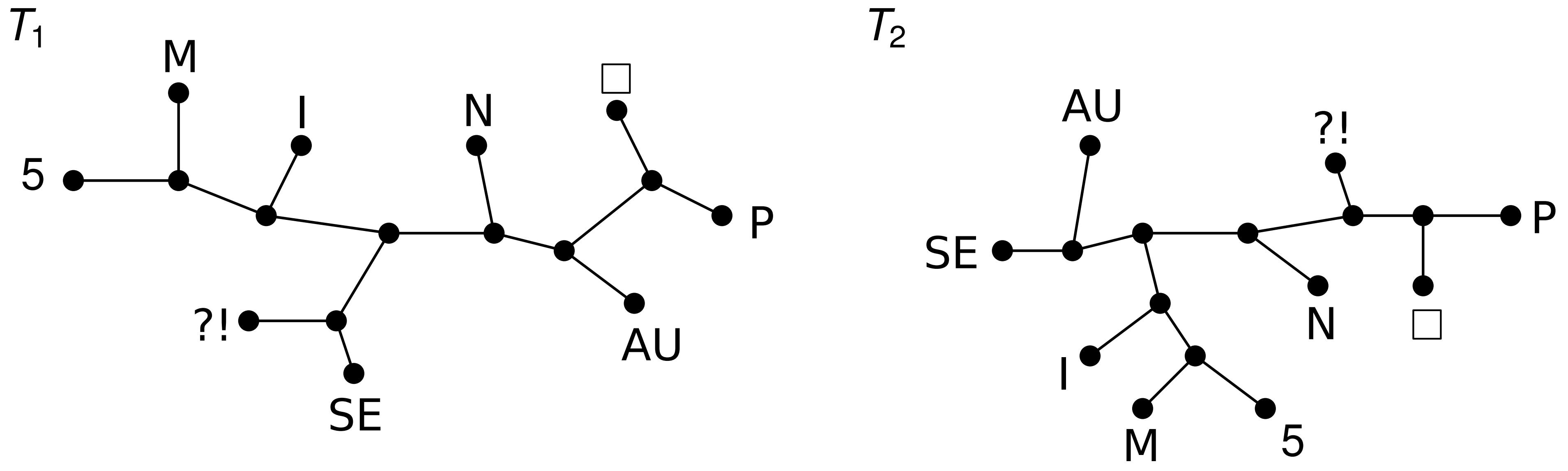
Beispiel

Löse diese Maximum Agreement Forest-Instanz



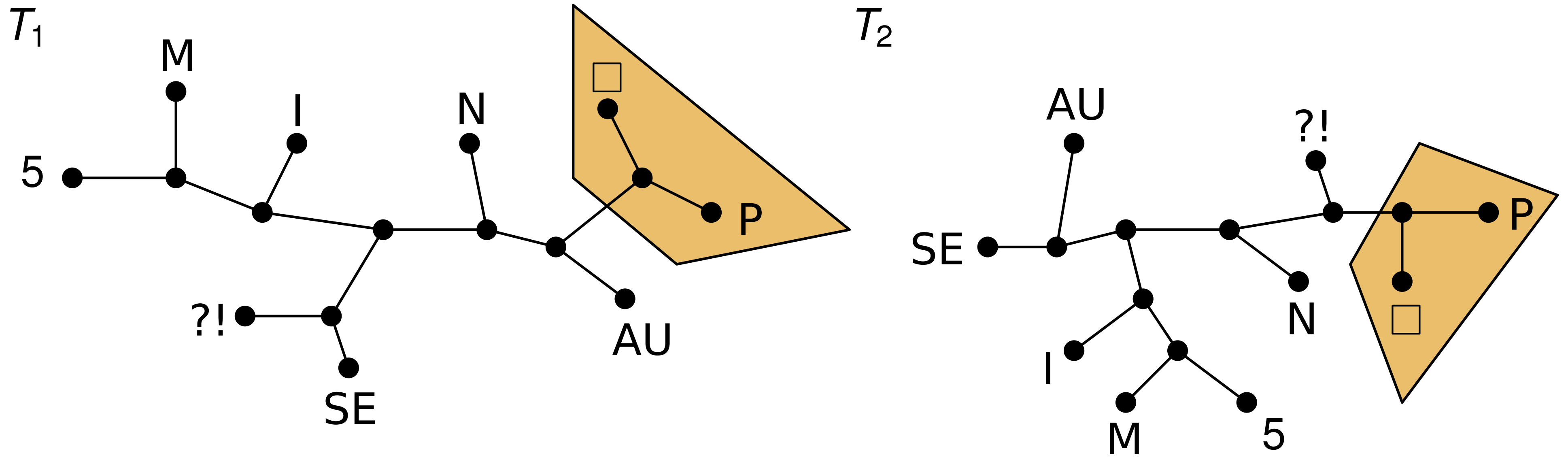
Beispiel

Löse diese Maximum Agreement Forest-Instanz



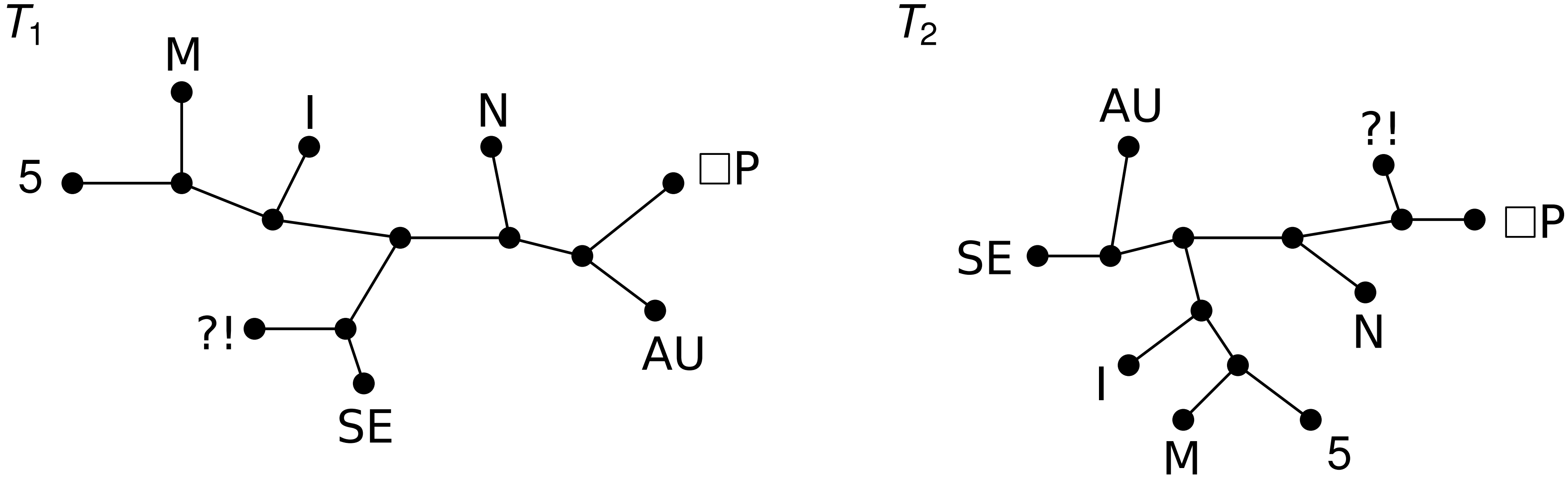
Beispiel

Löse diese Maximum Agreement Forest-Instanz



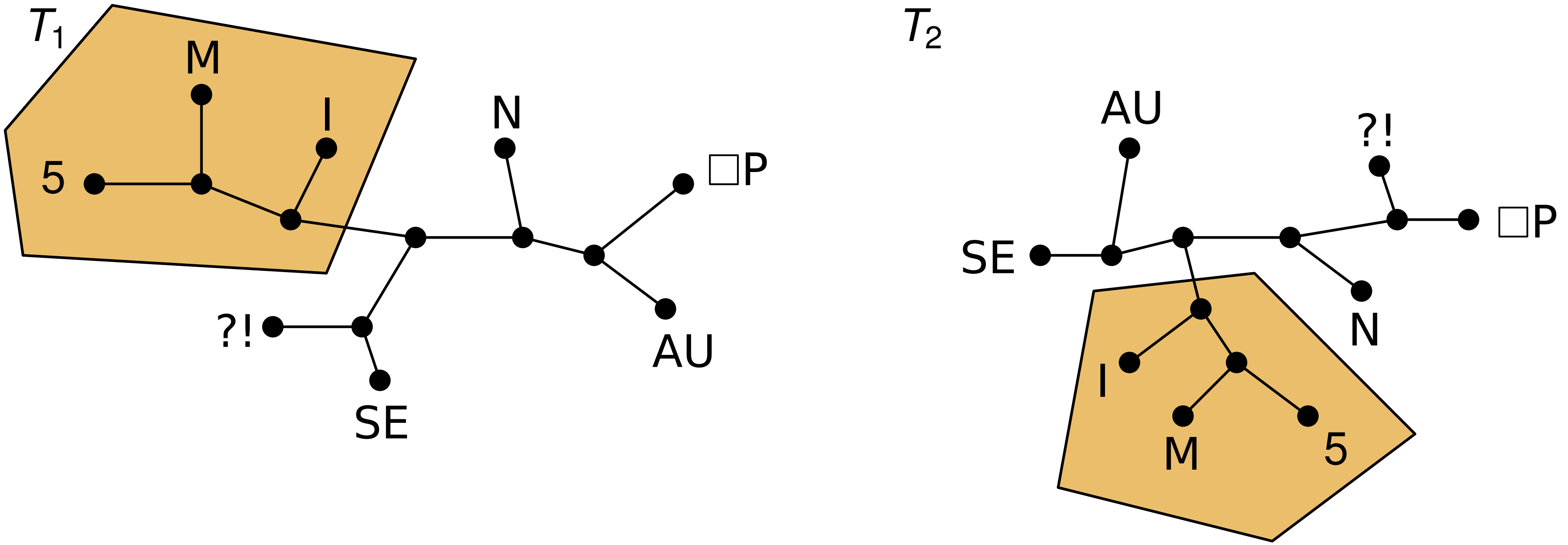
Beispiel

Löse diese Maximum Agreement Forest-Instanz



Beispiel

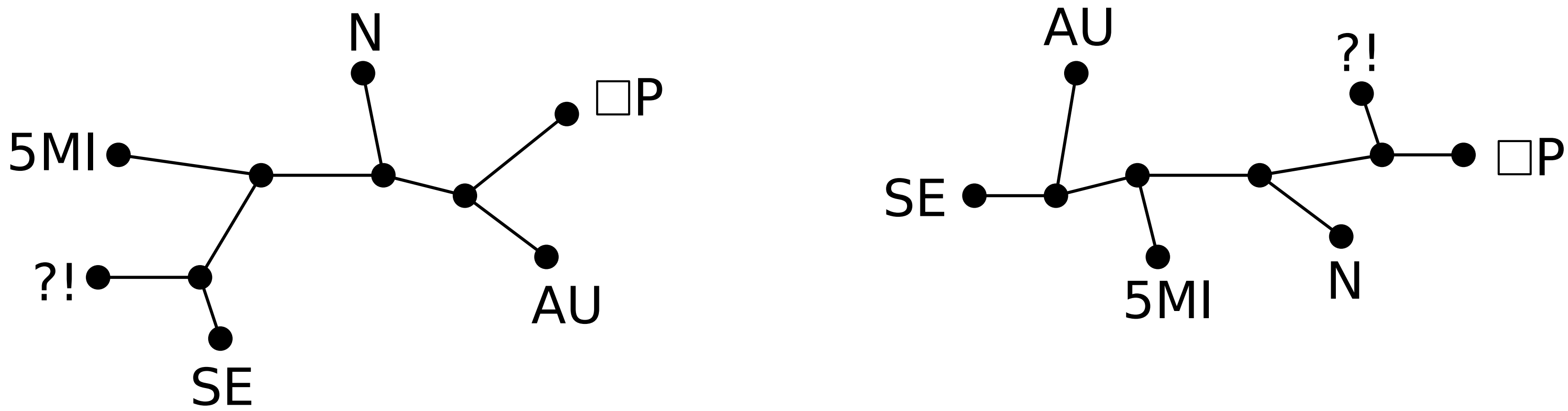
Löse diese Maximum Agreement Forest-Instanz



Beispiel

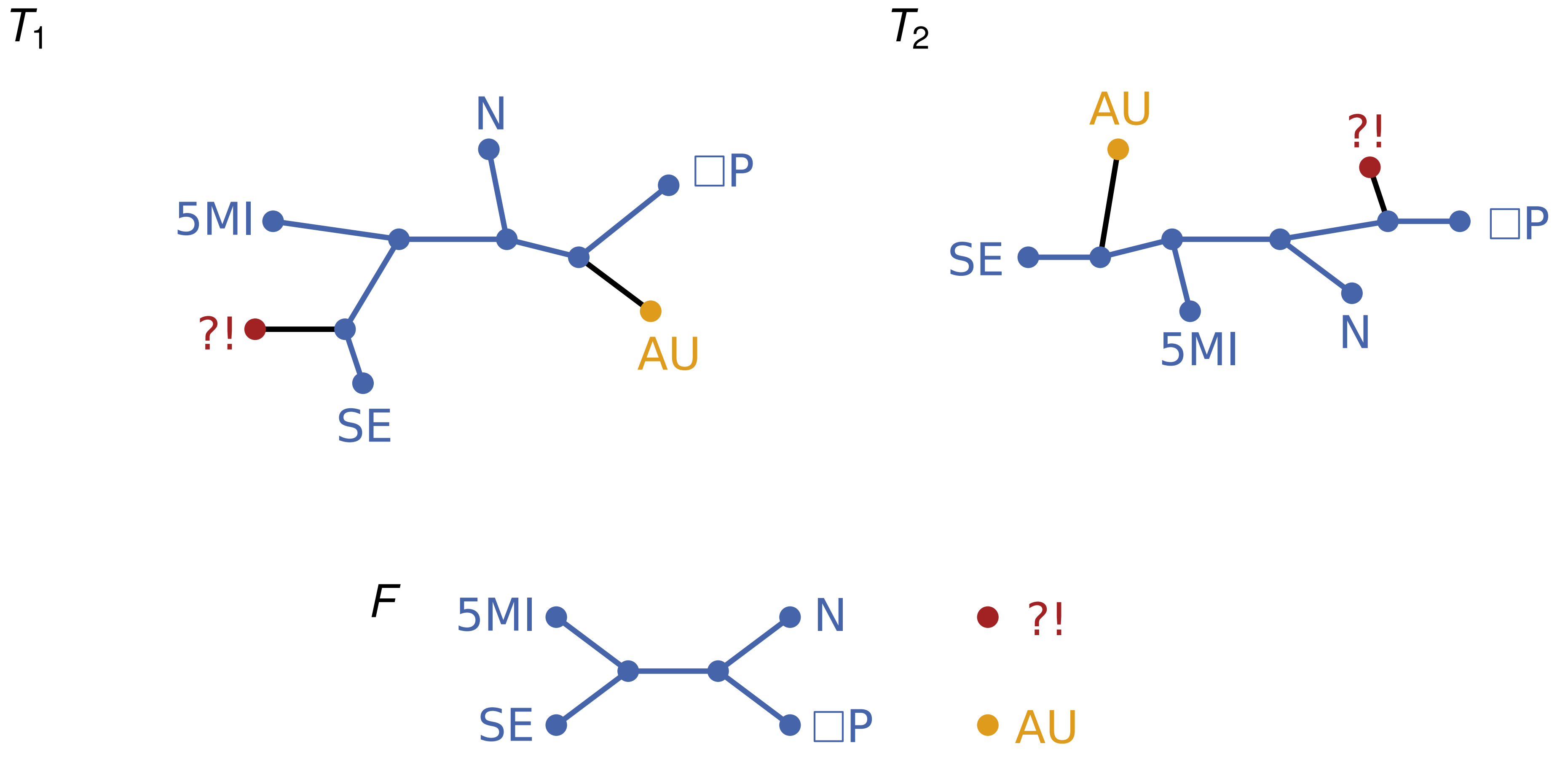
Löse diese Maximum Agreement Forest-Instanz

T_1 T_2



Beispiel

Löse diese Maximum Agreement Forest-Instanz



$\Rightarrow$ Lösung hat die Größe 3

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

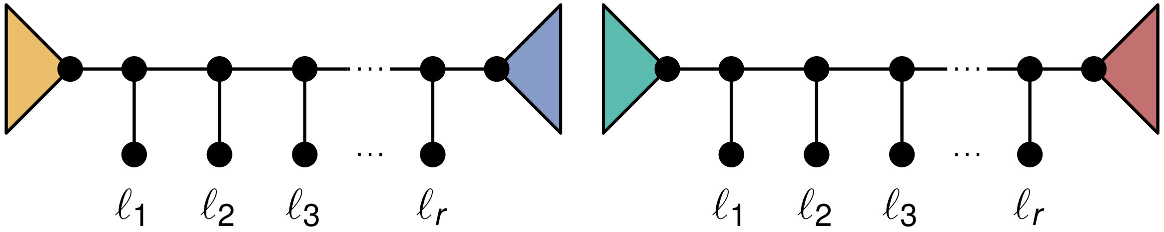
Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum

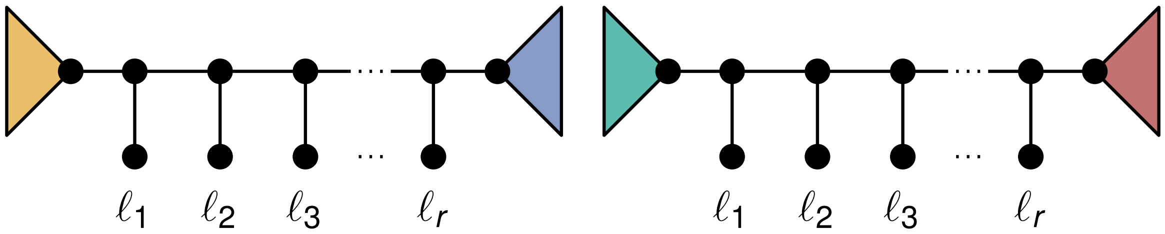


Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



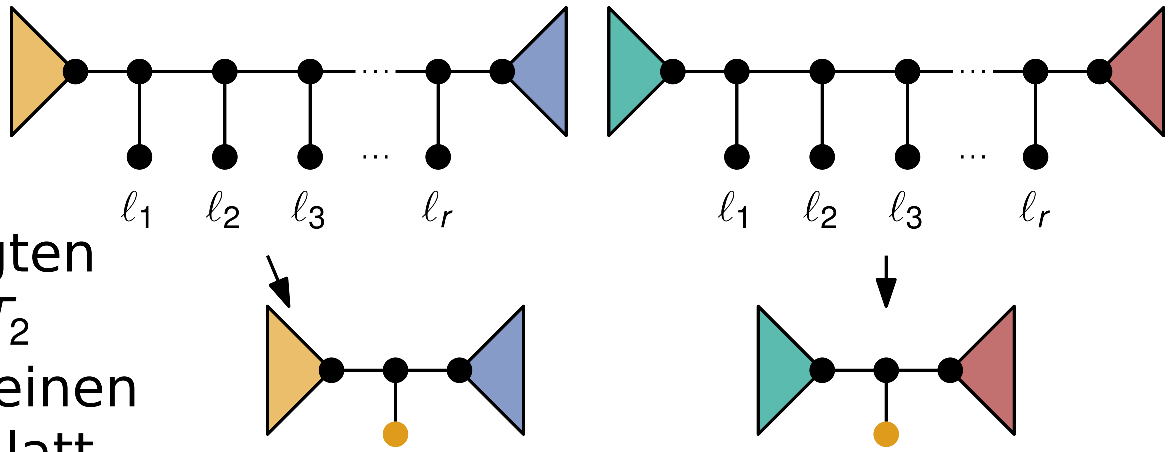
Reduktionsregel 2

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze den Pfad durch einen Knoten mit einem neuen Blatt

Ungünstige Baumstruktur 2

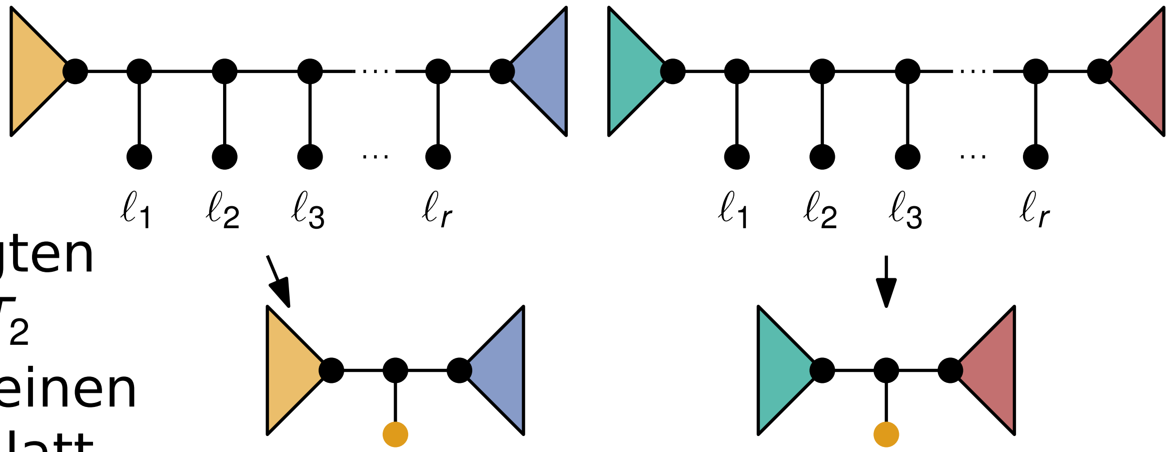
Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum

Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \ldots, \ell_r$ in T_1 und T_2
- ersetze den Pfad durch einen Knoten mit einem neuen Blatt



Ist diese Regel sicher?

Ungünstige Baumstruktur 2

Karlsruher Institut für Technologie

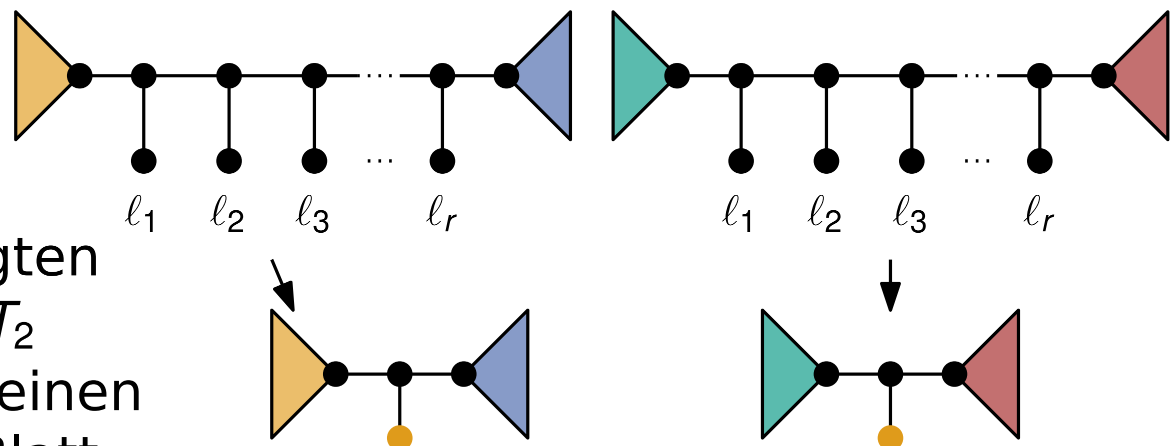
Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum

Reduktionsregel 2

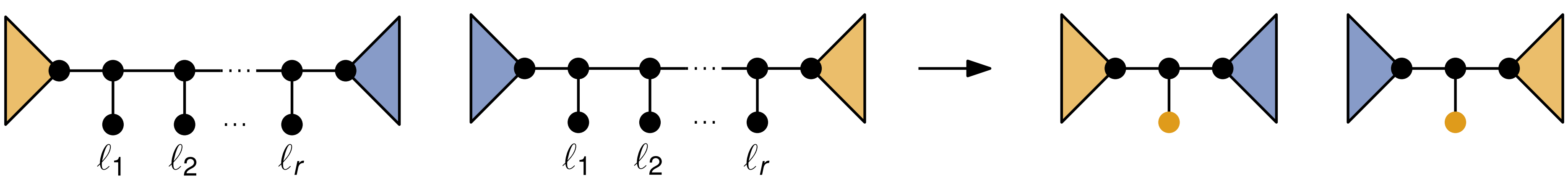
- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze den Pfad durch einen Knoten mit einem neuen Blatt



Ist diese Regel sicher?

Problem

- Anzahl notwendiger Bäume wird ggf. verringert



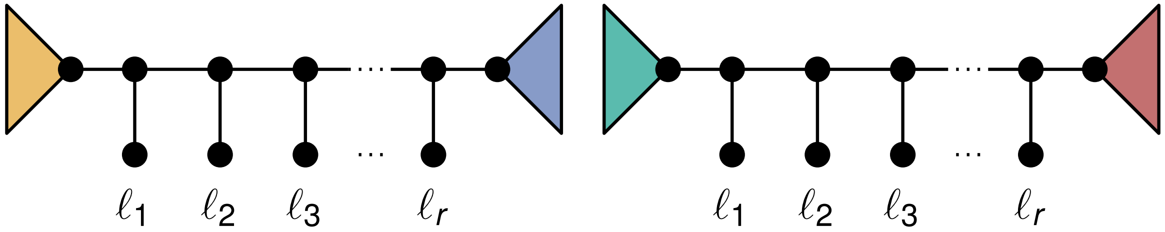
drei Bäume nötig — ein Baum genügt

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

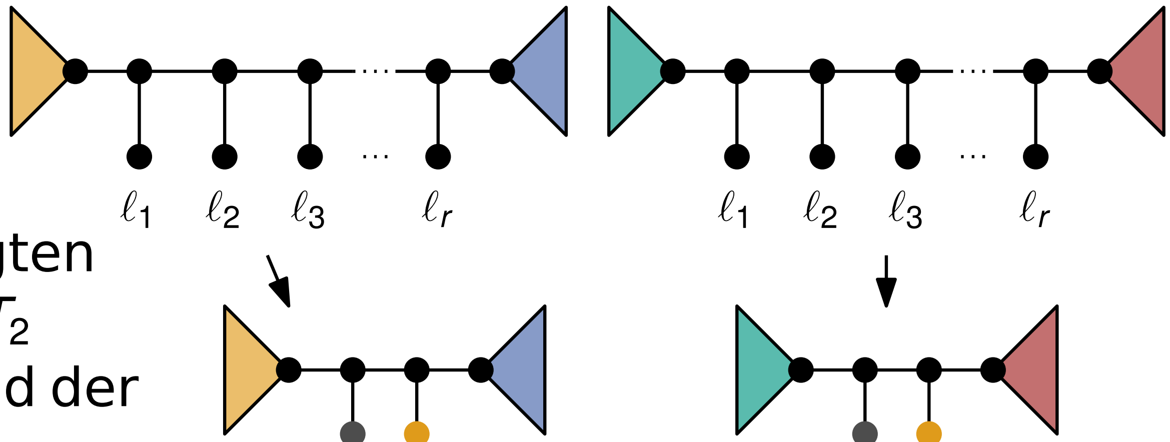
Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

(trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

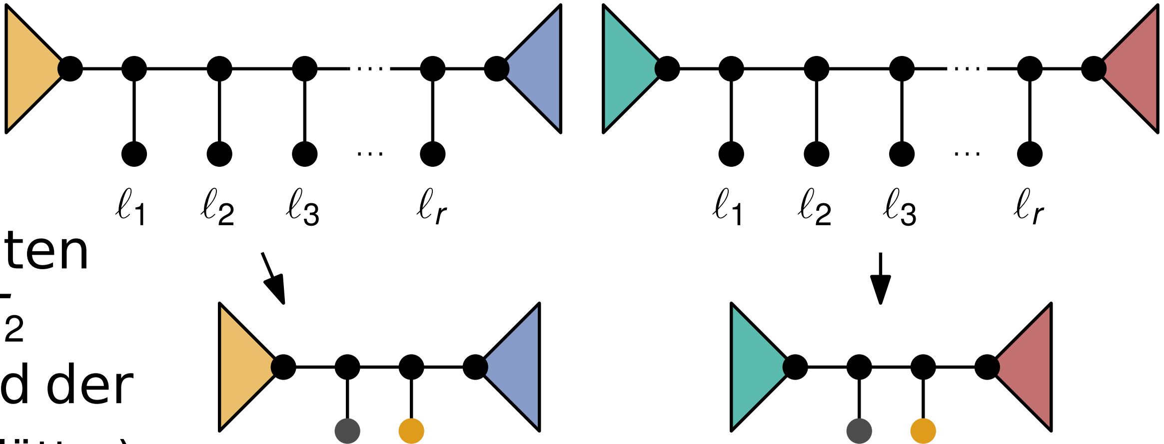
- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 2 (zwei neue Blätter)

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \ldots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 2 (zwei neue Blätter)

Ist diese Regel sicher?

Ungünstige Baumstruktur 2

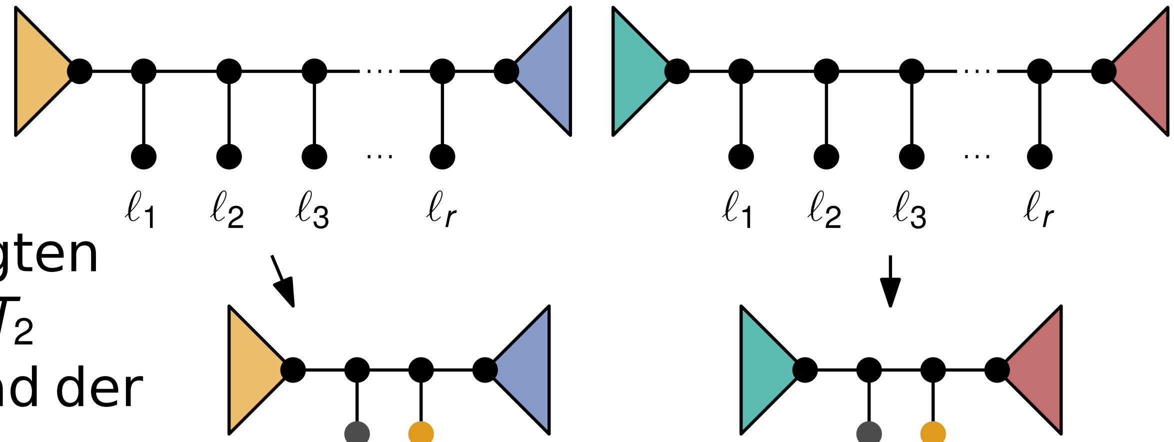
Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum

Reduktionsregel 2

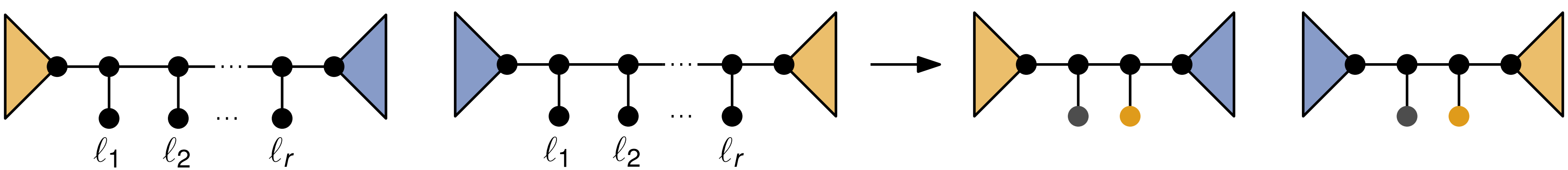
- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 2 (zwei neue Blätter)



Ist diese Regel sicher?

Problem

- Anzahl notwendiger Bäume wird ggf. verringert



drei Bäume nötig

zwei Bäume genügen

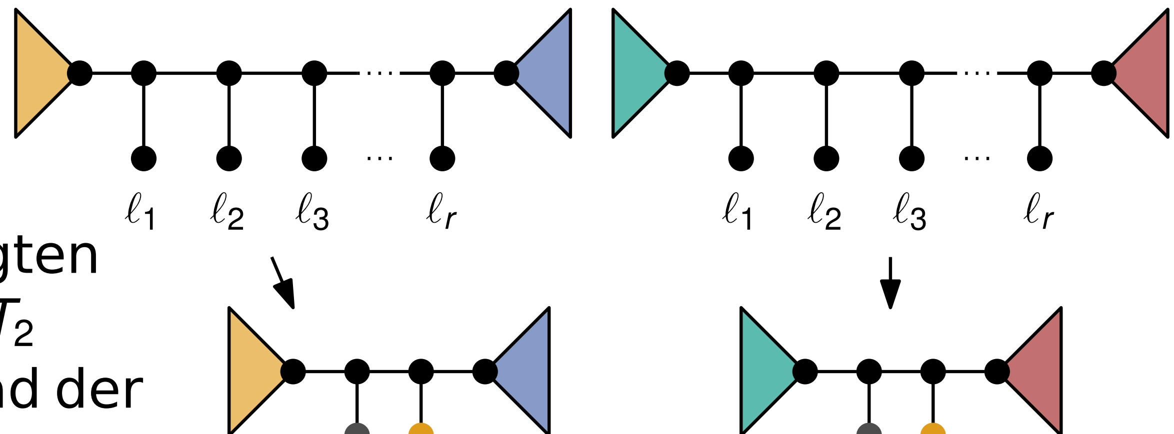
Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

(trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



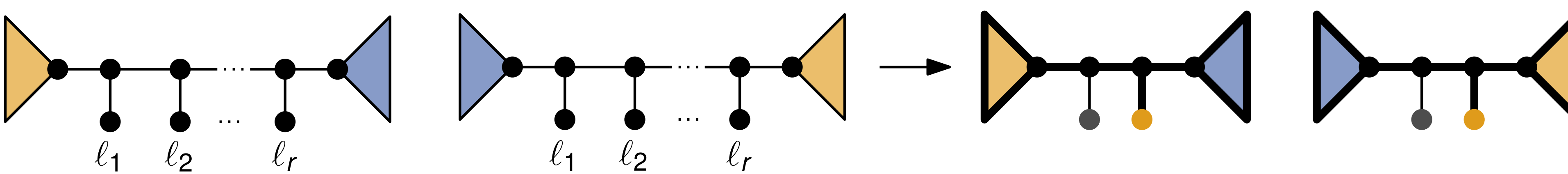
Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 2 (zwei neue Blätter)

Ist diese Regel sicher?

Problem

- Anzahl notwendiger Bäume wird ggf. verringert



drei Bäume nötig

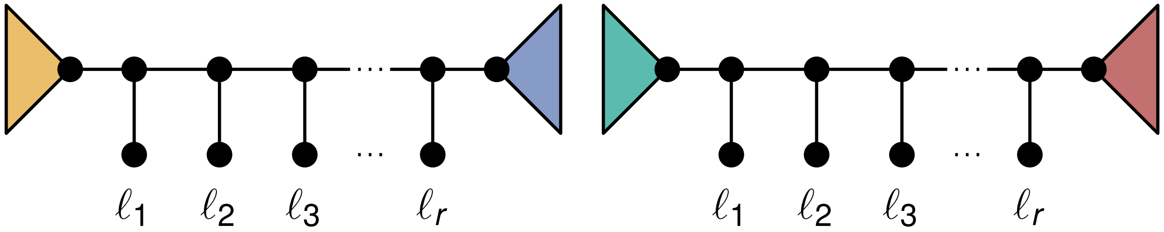
zwei Bäume genügen

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



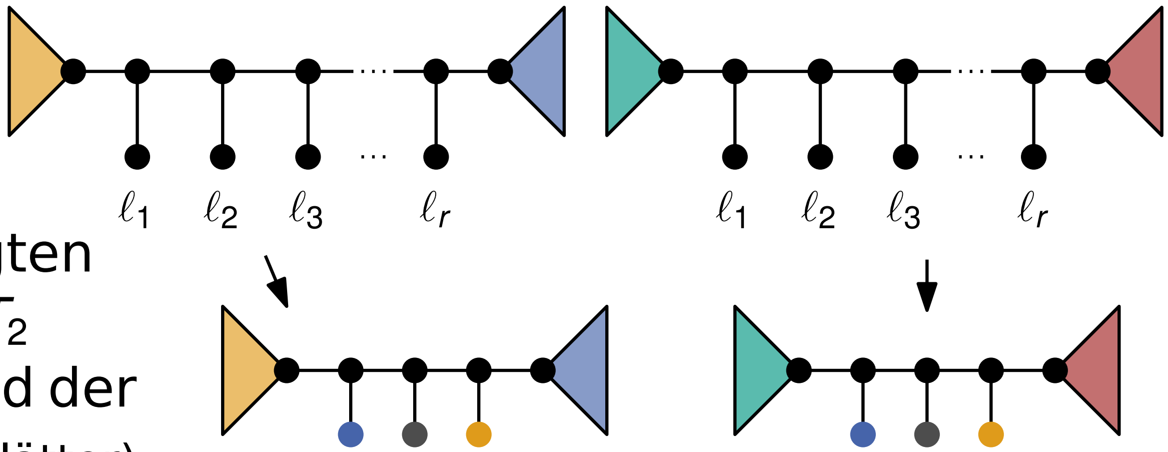
Reduktionsregel 2

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 3 (drei neue Blätter)

Ungünstige Baumstruktur 2

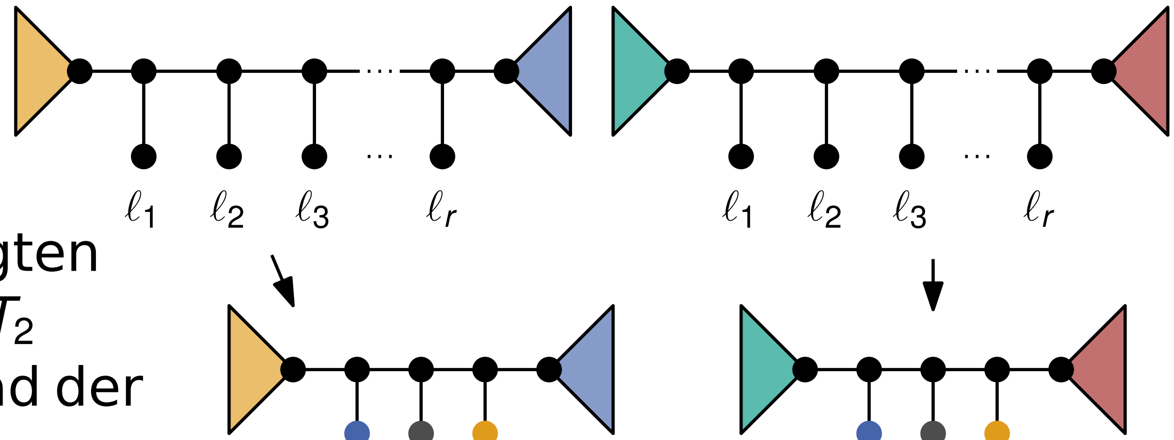
Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum

Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 3 (drei neue Blätter)



Lemma
Reduktionsregel 2 ist sicher und in polynomieller Zeit ausführbar.

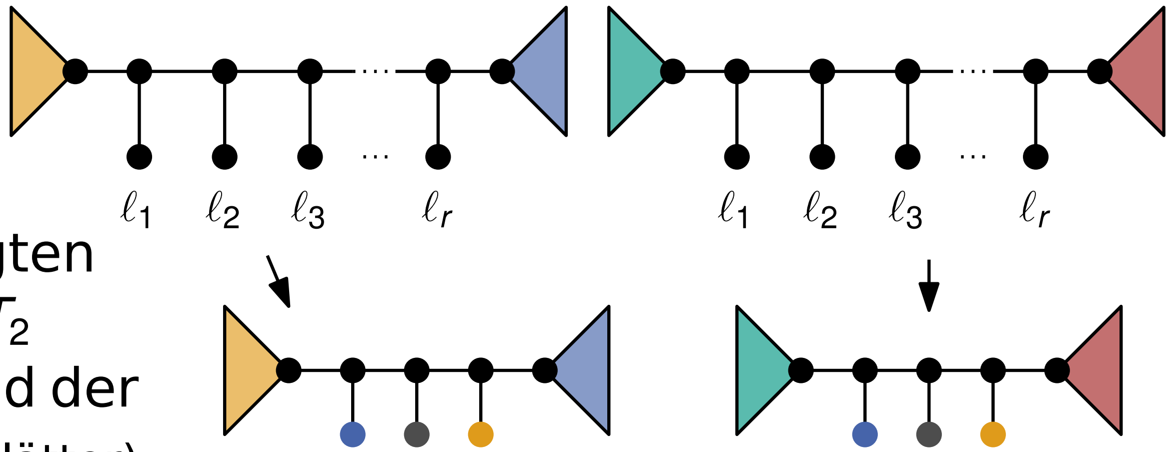
Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

(trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 3 (drei neue Blätter)

Lemma
Reduktionsregel 2 ist sicher und in polynomieller Zeit ausführbar.

Beweis

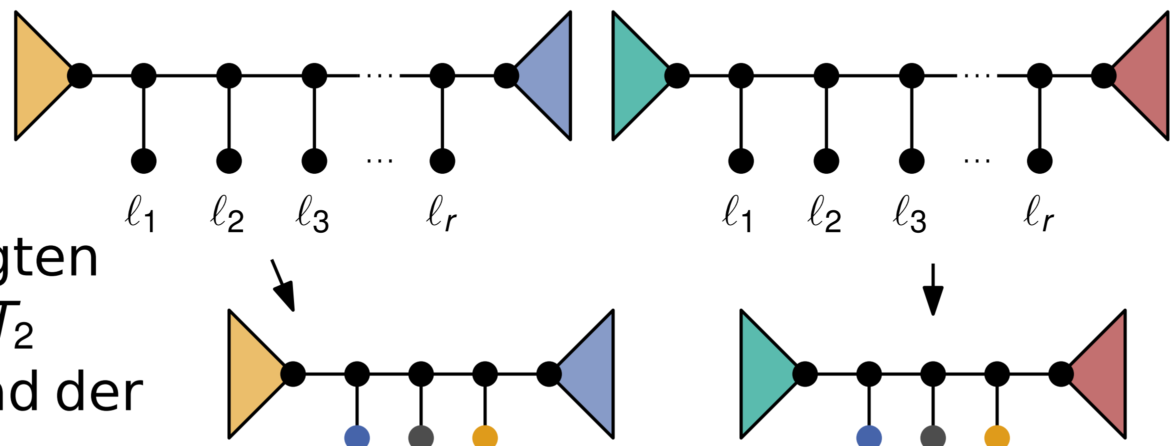
- Fallunterscheidung (nicht hier, aber nicht schwer)

Ungünstige Baumstruktur 2

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregel 1)

- viele Blätter in geteiltem Teilbaum



Reduktionsregel 2

- finde Pfad mit angehängten Blättern $\ell_1, \dots, \ell_r$ in T_1 und T_2
- ersetze ihn durch einen Pfad der Länge 3 (drei neue Blätter)

Lemma
Reduktionsregel 2 ist sicher und in polynomieller Zeit ausführbar.

Beweis

- Fallunterscheidung (nicht hier, aber nicht schwer)
- polynomielle Laufzeit: z.B. durch iteratives Verkürzen von Pfaden der Länge 4

Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

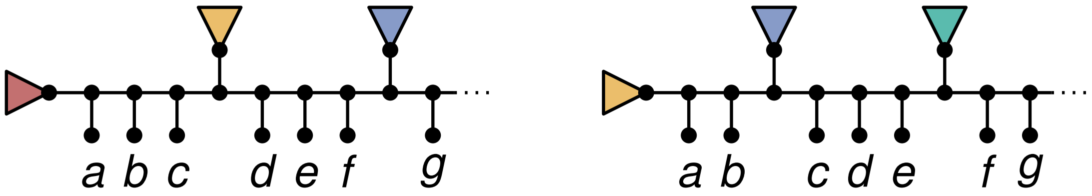
Gegenbeispiel (trotz Anwendung der Reduktionsregeln 1 und 2)

Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregeln 1 und 2)

- viele Blätter ($a, b, c, \ldots$) in geteiltem Teilbaum



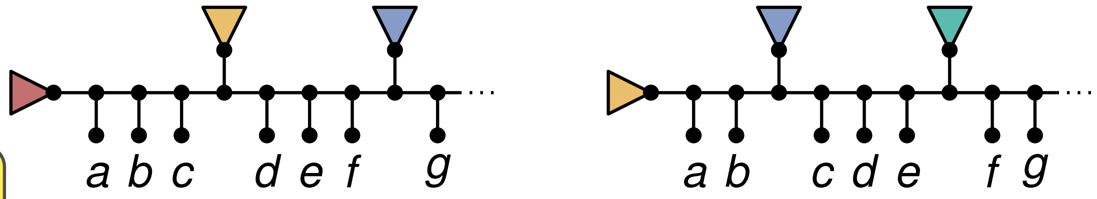
Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel

(trotz Anwendung der Reduktionsregeln 1 und 2)

- viele Blätter ($a, b, c, \ldots$) in geteiltem Teilbaum



Ist das wirklich ein Problem?

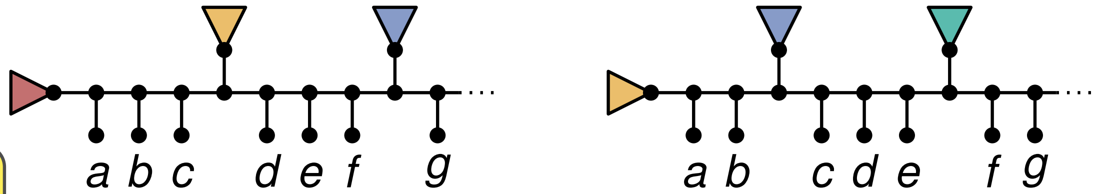
Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält $\underbrace{\text{nur wenige}}_{}$ Blätter. amortisiert

Gegenbeispiel

(trotz Anwendung der Reduktionsregeln 1 und 2)

- viele Blätter ($a, b, c, \ldots$) in geteiltem Teilbaum



Ist das wirklich ein Problem?

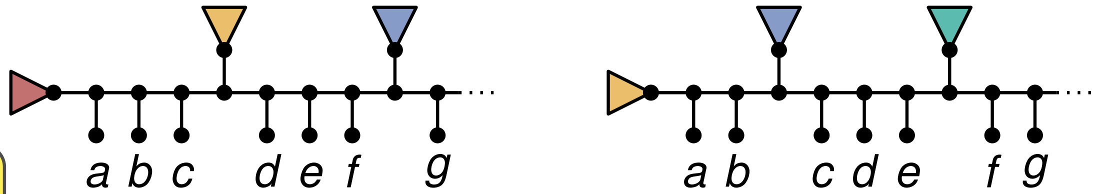
Amortisierte Sichtweise

- nach je drei Blättern kommen sich unterscheidende Teilbäume

Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält amortisiert nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregeln 1 und 2)



- viele Blätter ($a, b, c, \dots$) in geteiltem Teilbaum

Ist das wirklich ein Problem?

Amortisierte Sichtweise

- nach je drei Blättern kommen sich unterscheidende Teilbäume
- jeder dieser Teilbäume sorgt für (mindestens) einen eigenen Baum im Agreement Forest (wenn $a, b, c, d, \dots$ alle im gleichen Baum liegen)

Ungünstige Baumstruktur 3

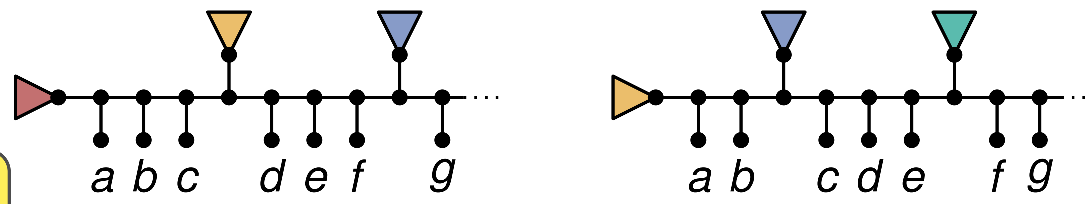
Behauptung

amortisiert

Jeder Baum in einem Agreement Forest enthält nur wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregeln 1 und 2)

- viele Blätter ($a, b, c, \ldots$) in geteiltem Teilbaum



Ist das wirklich ein Problem?

Amortisierte Sichtweise

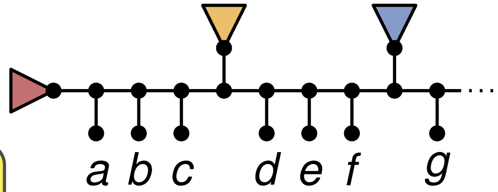
- nach je drei Blättern kommen sich unterscheidende Teilbäume
- jeder dieser Teilbäume sorgt für (mindestens) einen eigenen Baum im Agreement Forest (wenn $a, b, c, d, \ldots$ alle im gleichen Baum liegen)
- wenige Bäume im Agreement Forest $\Rightarrow$ insgesamt wenige Blätter

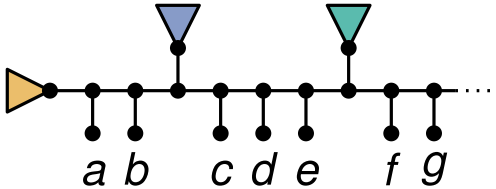
Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält $\underbrace{\text{nur}}^{\text{amortisiert}}$ wenige Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregeln 1 und 2)

- viele Blätter ($a, b, c, \dots$) in geteiltem Teilbaum





Ist das wirklich ein Problem?

Amortisierte Sichtweise

- nach je drei Blättern kommen sich unterscheidende Teilbäume
- jeder dieser Teilbäume sorgt für (mindestens) einen eigenen Baum im Agreement Forest (wenn $a, b, c, d, \dots$ alle im gleichen Baum liegen)
- wenige Bäume im Agreement Forest $\Rightarrow$ insgesamt wenige Blätter

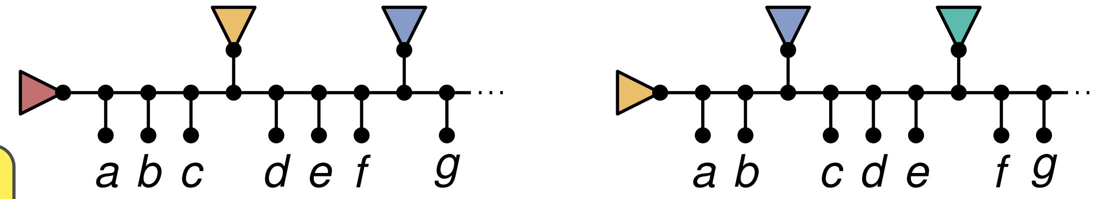
Sicher, dass es keine anderen ungünstigen Strukturen geben kann?

Ungünstige Baumstruktur 3

Behauptung
Jeder Baum in einem Agreement Forest enthält $\overbrace{\text{nur wenige}}^{\text{amortisiert}}$ Blätter.

Gegenbeispiel (trotz Anwendung der Reduktionsregeln 1 und 2)

- viele Blätter ($a, b, c, \dots$) in geteiltem Teilbaum



Ist das wirklich ein Problem?

Amortisierte Sichtweise

- nach je drei Blättern kommen sich unterscheidende Teilbäume
- jeder dieser Teilbäume sorgt für (mindestens) einen eigenen Baum im Agreement Forest (wenn $a, b, c, d, \dots$ alle im gleichen Baum liegen)
- wenige Bäume im Agreement Forest $\Rightarrow$ insgesamt wenige Blätter

Sicher, dass es keine anderen ungünstigen Strukturen geben kann?

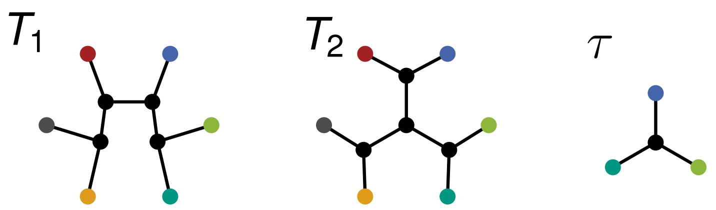
Lemma
In einer lösbaren Instanz liefern die Reduktionsregeln 1 und 2 eine äquivalente Instanz mit maximal ck Blättern (für eine kleine Konstante c).

Grober Plan

Notation

- betrachte einen der Bäume τ in einem Agreement Forest F
- kontrahiere $T_1(L(\tau))$ in T_1 zu einem einzelnen Knoten
- bezeichne den Grad dieses Knotens mit $\deg_1(\tau)$ (analog: $\deg_2(\tau)$)

Beispiel



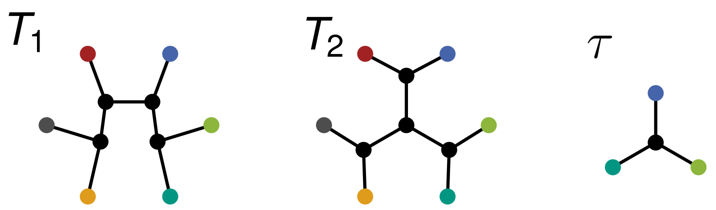
- $\deg_1(\tau) = 1$
- $\deg_2(\tau) = 2$

Grober Plan

Notation

- betrachte einen der Bäume τ in einem Agreement Forest F
- kontrahiere $T_1(L(\tau))$ in T_1 zu einem einzelnen Knoten
- bezeichne den Grad dieses Knotens mit $\deg_1(\tau)$ (analog: $\deg_2(\tau)$)

Beispiel



- $\deg_1(\tau) = 1$
- $\deg_2(\tau) = 2$

Amortisierte Sichtweise

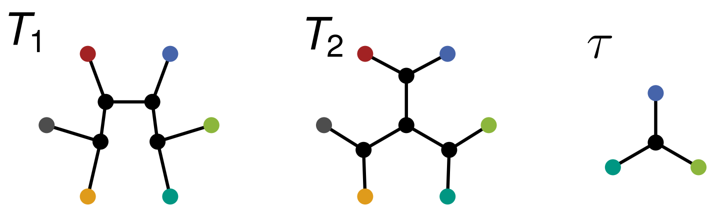
(wie eben, nur etwas formaler)

Grober Plan

Notation

- betrachte einen der Bäume τ in einem Agreement Forest F
- kontrahiere $T_1(L(\tau))$ in T_1 zu einem einzelnen Knoten
- bezeichne den Grad dieses Knotens mit $\deg_1(\tau)$ (analog: $\deg_2(\tau)$)

Beispiel



- $\deg_1(\tau) = 1$
- $\deg_2(\tau) = 2$

Amortisierte Sichtweise

(wie eben, nur etwas formaler)

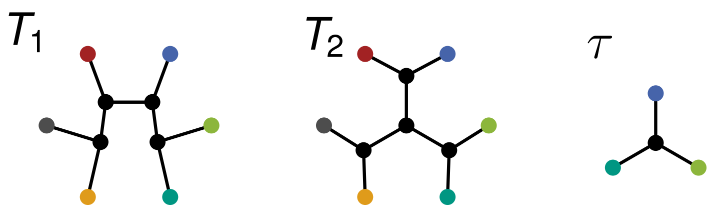
- τ zerlegt T_i in $\deg_i(\tau)$ viele Teilbäume (für $i \in \{1, 2\}$)

Grober Plan

Notation

- betrachte einen der Bäume τ in einem Agreement Forest F
- kontrahiere $T_1(L(\tau))$ in T_1 zu einem einzelnen Knoten
- bezeichne den Grad dieses Knotens mit $\deg_1(\tau)$ (analog: $\deg_2(\tau)$)

Beispiel



- $\deg_1(\tau) = 1$
- $\deg_2(\tau) = 2$

Amortisierte Sichtweise (wie eben, nur etwas formaler)

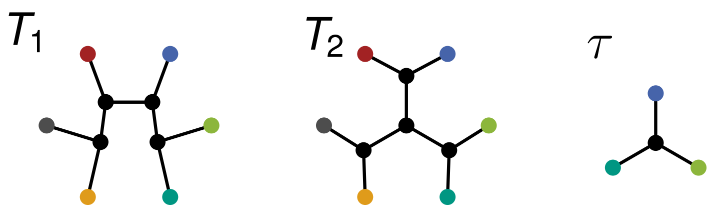
- τ zerlegt T_i in $\deg_i(\tau)$ viele Teilbäume (für $i \in \{1, 2\}$)
- es gibt also mindestens $\deg_i(\tau)$ weitere Bäume in F

Grober Plan

Notation

- betrachte einen der Bäume τ in einem Agreement Forest F
- kontrahiere $T_1(L(\tau))$ in T_1 zu einem einzelnen Knoten
- bezeichne den Grad dieses Knotens mit $\deg_1(\tau)$ (analog: $\deg_2(\tau)$)

Beispiel



- $\deg_1(\tau) = 1$
- $\deg_2(\tau) = 2$

Amortisierte Sichtweise

(wie eben, nur etwas formaler)

- τ zerlegt T_i in $\deg_i(\tau)$ viele Teilbäume (für $i \in \{1, 2\}$)
- es gibt also mindestens $\deg_i(\tau)$ weitere Bäume in F

Lemma

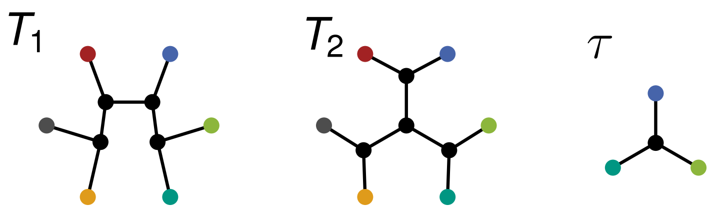
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Grober Plan

Notation

- betrachte einen der Bäume τ in einem Agreement Forest F
- kontrahiere $T_1(L(\tau))$ in T_1 zu einem einzelnen Knoten
- bezeichne den Grad dieses Knotens mit $\deg_1(\tau)$ (analog: $\deg_2(\tau)$)

Beispiel



- $\deg_1(\tau) = 1$
- $\deg_2(\tau) = 2$

Amortisierte Sichtweise

(wie eben, nur etwas formaler)

- τ zerlegt T_i in $\deg_i(\tau)$ viele Teilbäume (für $i \in \{1, 2\}$)
- es gibt also mindestens $\deg_i(\tau)$ weitere Bäume in F

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

Summe der Grade

Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

Beweis

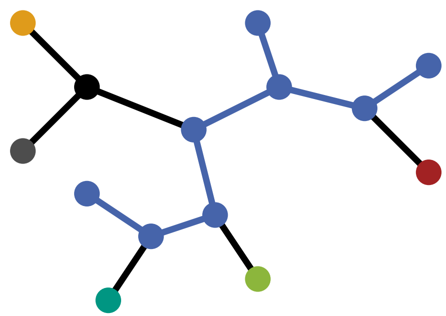
Summe der Grade

Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

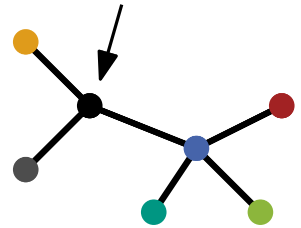
Beweis

- kontrahiere jedes $\tau \in F$ zu einzelnem Knoten in $T_i \to$ neuer Baum T_i'
- jeder Knoten in T_i' gehört zu einem der k Bäume/ist *nicht kontrahiert*

T_i (Bäume aus F bunt)



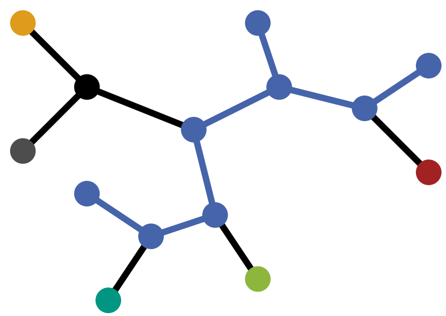
T_i' nicht kontrahiert



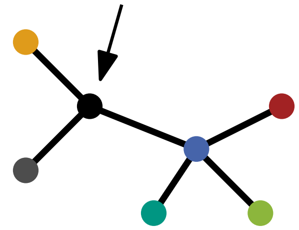
Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

Beweis
- kontrahiere jedes $\tau \in F$ zu einzelnem Knoten in $T_i \rightarrow$ neuer Baum T_i'
- jeder Knoten in T_i' gehört zu einem der k Bäume/ist *nicht kontrahiert*

T_i (Bäume aus F bunt)



T_i'



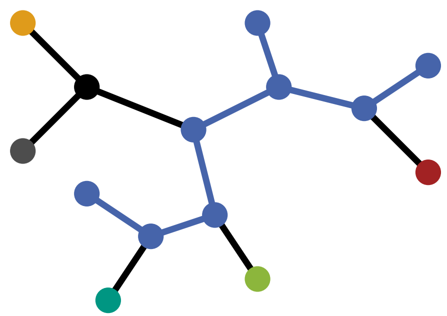
- nicht kontrahierte Knoten haben Grad 3
- n_3 = Anzahl dieser Knoten

Summe der Grade

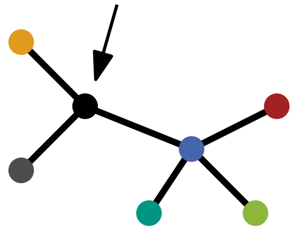
Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

Beweis
- kontrahiere jedes $\tau \in F$ zu einzelnem Knoten in T_i $\rightarrow$ neuer Baum T_i'
- jeder Knoten in T_i' gehört zu einem der k Bäume/ist *nicht kontrahiert*

T_i (Bäume aus F bunt)



T_i' nicht kontrahiert



- nicht kontrahierte Knoten haben Grad 3
- n_3 = Anzahl dieser Knoten
- m' = Anzahl Kanten in T_i'
- n' = Anzahl Knoten in T_i'

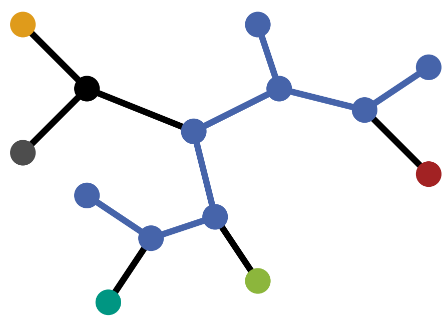
Summe der Grade

Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

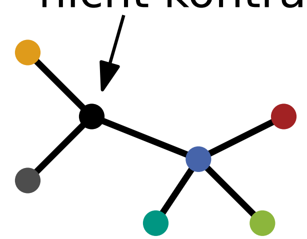
Beweis

- kontrahiere jedes $\tau \in F$ zu einzelnem Knoten in $T_i \to$ neuer Baum T_i'
- jeder Knoten in T_i' gehört zu einem der k Bäume/ist *nicht kontrahiert*

T_i (Bäume aus F bunt)



T_i' nicht kontrahiert



- nicht kontrahierte Knoten haben Grad 3
- n_3 = Anzahl dieser Knoten
- m' = Anzahl Kanten in T_i'
- n' = Anzahl Knoten in T_i'
- es gilt (handshaking): $\sum_{\tau \in F} \deg_i(\tau) + 3n_3 = 2m'$

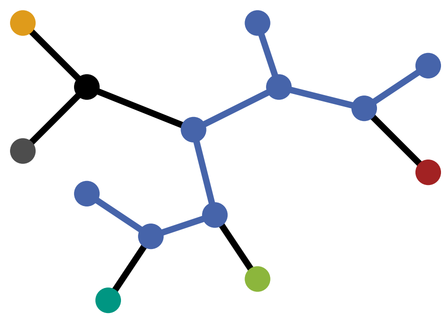
Summe der Grade

Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

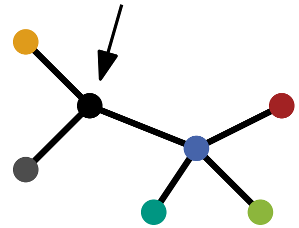
Beweis

- kontrahiere jedes $\tau \in F$ zu einzelnem Knoten in $T_i \to$ neuer Baum T_i'
- jeder Knoten in T_i' gehört zu einem der k Bäume/ist *nicht kontrahiert*

T_i (Bäume aus F bunt)



T_i' nicht kontrahiert



- nicht kontrahierte Knoten haben Grad 3
- n_3 = Anzahl dieser Knoten
- m' = Anzahl Kanten in T_i'
- n' = Anzahl Knoten in T_i'

- es gilt (handshaking): $\sum_{\tau \in F} \deg_i(\tau) + 3n_3 = 2m'$
- außerdem (T_i' ist Baum): $n' = k + n_3 = m' + 1$

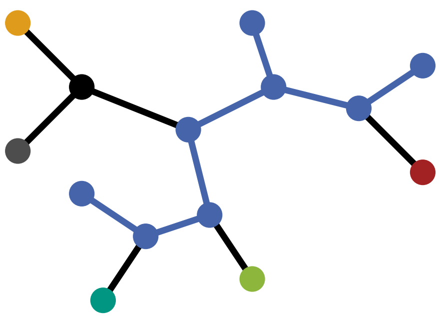
Summe der Grade

Lemma
Wenn F aus k Bäumen besteht, dann gilt $\sum_{\tau \in F} \deg_i(\tau) \leq 2k - 2$.

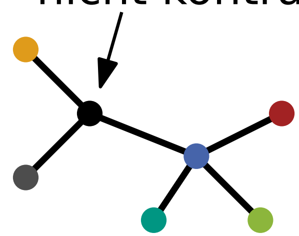
Beweis

- kontrahiere jedes $\tau \in F$ zu einzelnem Knoten in $T_i \to$ neuer Baum T_i'
- jeder Knoten in T_i' gehört zu einem der k Bäume/ist *nicht kontrahiert*

T_i (Bäume aus F bunt)



T_i' nicht kontrahiert



- nicht kontrahierte Knoten haben Grad 3
- n_3 = Anzahl dieser Knoten
- m' = Anzahl Kanten in T_i'
- n' = Anzahl Knoten in T_i'

- es gilt (handshaking): $\sum_{\tau \in F} \deg_i(\tau) + 3n_3 = 2m'$
- außerdem (T_i' ist Baum): $n' = k + n_3 = m' + 1$

$$\Rightarrow \sum_{\tau \in F} \deg_i(\tau) = 2(k + n_3 - 1) - 3n_3 \leq 2k - 2$$

Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

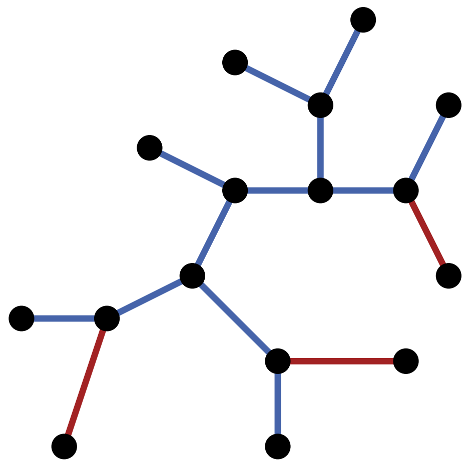
Beweis

Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2

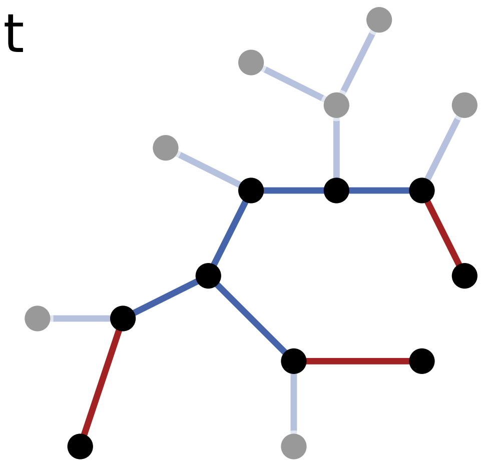


Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält

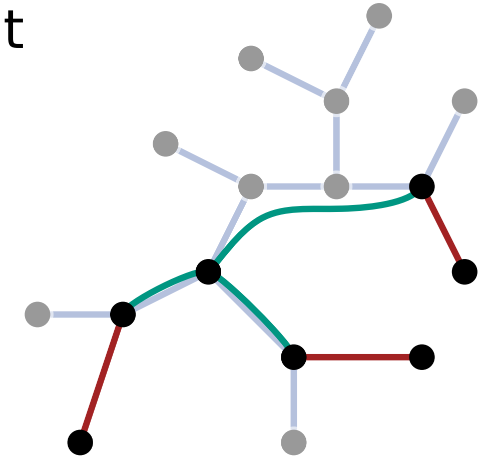


Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält
  - kontrahiere blaue Pfade zu grünen Kanten

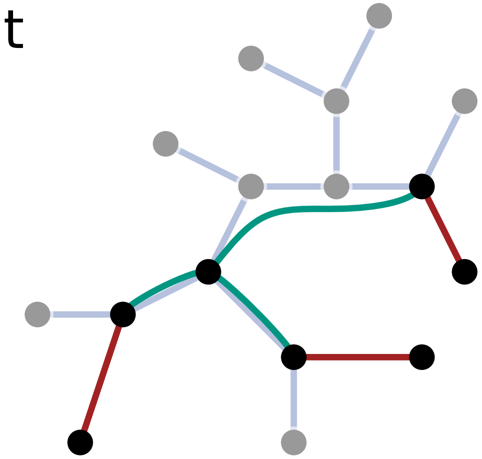


Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält
  - kontrahiere blaue Pfade zu grünen Kanten



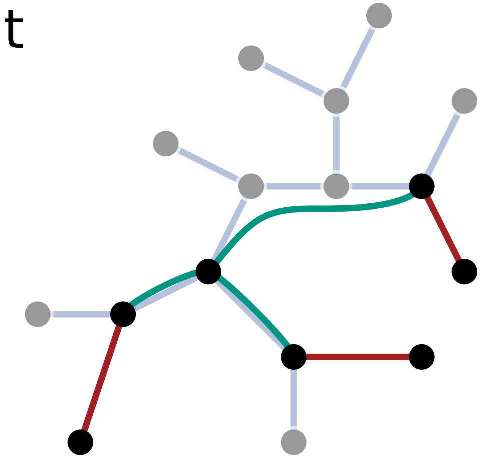
Wie viele Blätter hat τ?

Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält
  - kontrahiere blaue Pfade zu grünen Kanten



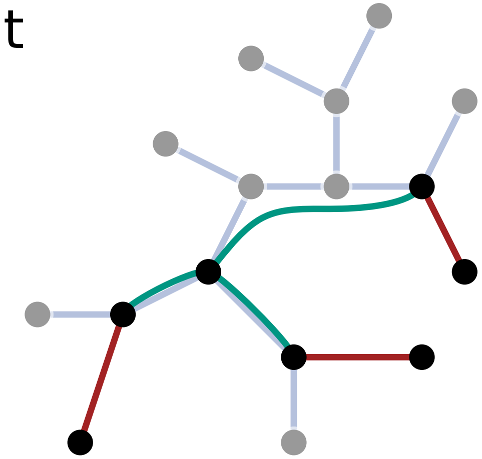
Wie viele Blätter hat τ?

- nur eins pro Knoten mit Grad 2 in τ' (Regel 1)

Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis
- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält
  - kontrahiere blaue Pfade zu grünen Kanten



Wie viele Blätter hat τ?
- nur eins pro Knoten mit Grad 2 in τ' (Regel 1)
- nur eins pro Knoten mit Grad 1 in τ' (Regel 1)

Wenige Blätter pro Baum

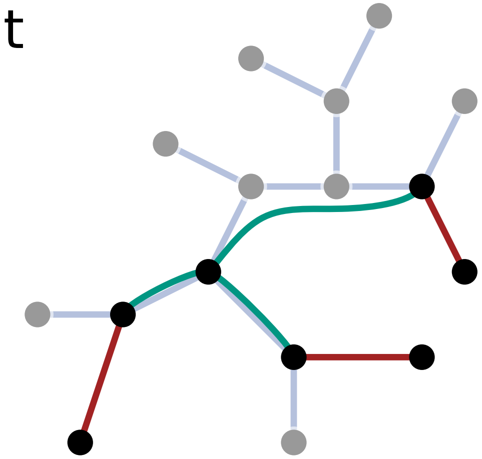
Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält
  - kontrahiere blaue Pfade zu grünen Kanten

Wie viele Blätter hat τ?

- nur eins pro Knoten mit Grad 2 in τ' (Regel 1)
- nur eins pro Knoten mit Grad 1 in τ' (Regel 1)
- nur drei pro grüne Kante in τ' (Regel 2)

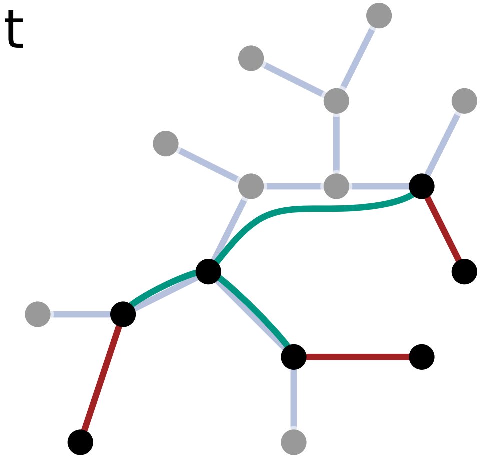


Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

- betrachte, wie τ in den beiden Bäumen T_1 bzw. T_2 liegt
- blaue Kanten: Kanten in T_1 und T_2; rote Kanten: Pfade in T_1 oder T_2
- konstruiere τ':
  - minimaler Teilbaum, der alle roten Kanten enthält
  - kontrahiere blaue Pfade zu grünen Kanten



Wie viele Blätter hat τ?

- nur eins pro Knoten mit Grad 2 in τ' (Regel 1)
- nur eins pro Knoten mit Grad 1 in τ' (Regel 1)
- nur drei pro grüne Kante in τ' (Regel 2)

Zeige: n_1, n_2, m_g sind nicht viel größer als m_r
$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

Wenige Blätter pro Baum

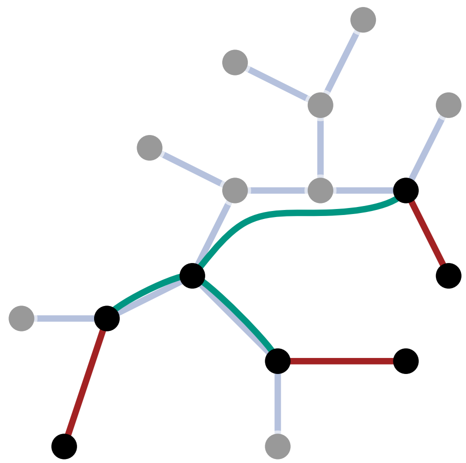
Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten



Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

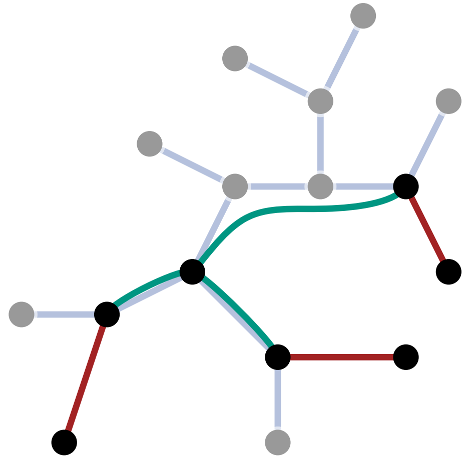
Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$

(Ausnahme: τ besteht aus einer Kante)



Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

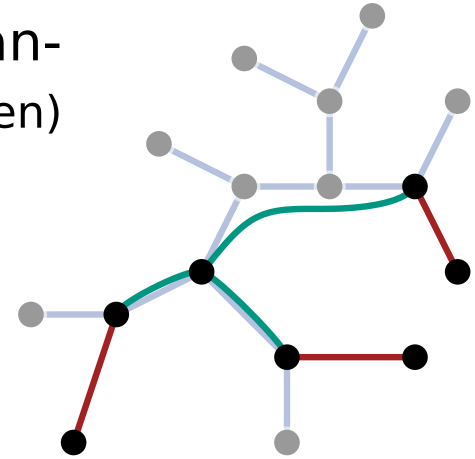
Beweis

Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)



Wenige Blätter pro Baum

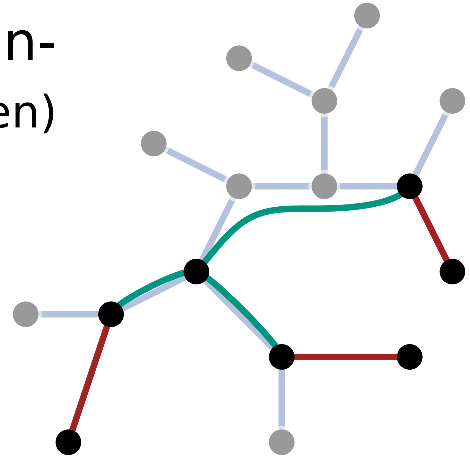
Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis
Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)
- Standardtrick in Bäumen – Kanten zählen:



Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

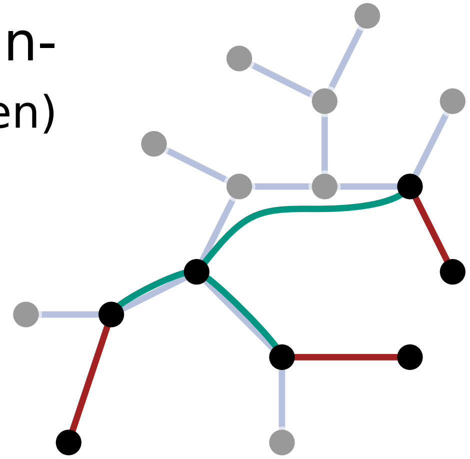
Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)
- Standardtrick in Bäumen – Kanten zählen:

$$m_r + m_g = n_1 + n_2 + n_3 - 1$$



Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

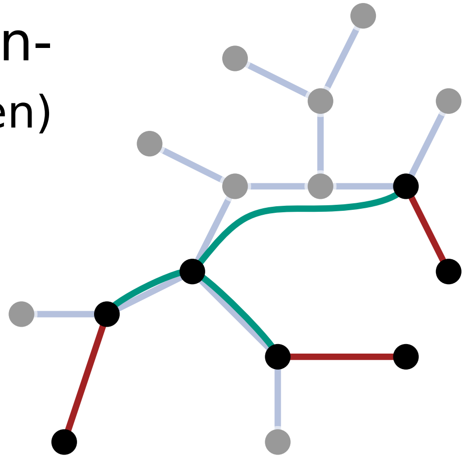
Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)
- Standardtrick in Bäumen – Kanten zählen:

$$m_r + m_g = n_1 + n_2 + n_3 - 1$$
$$2(m_r + m_g) = n_1 + 2n_2 + 3n_3$$



Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

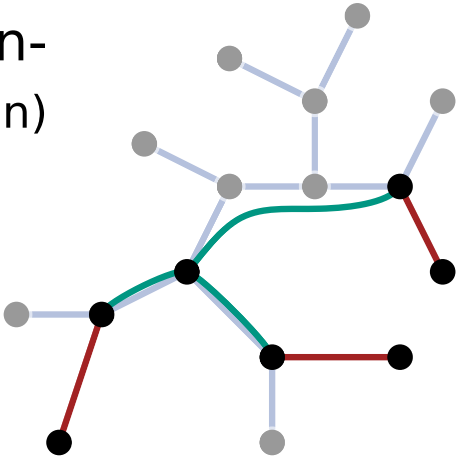
Beweis

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

Zeige: n_1, n_2, m_g sind nicht viel größer als m_r
$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)
- Standardtrick in Bäumen – Kanten zählen:

$$m_r + m_g = n_1 + n_2 + n_3 - 1 \quad \times 3$$
$$2(m_r + m_g) = n_1 + 2n_2 + 3n_3 \quad -1$$
$$m_r + m_g = 2n_1 + n_2 - 3$$



Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

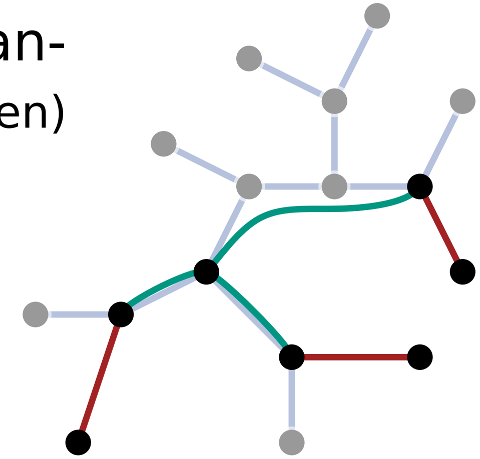
Beweis
Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)
- Standardtrick in Bäumen – Kanten zählen:

$$
\begin{aligned}
m_r + m_g &= n_1 + n_2 + n_3 - 1 && \times 3 \\
2(m_r + m_g) &= n_1 + 2n_2 + 3n_3 && -1 \\
m_r + m_g &= 2n_1 + n_2 - 3
\end{aligned}
$$



- also: $m_g \leq 2n_1 + n_2 - m_r \leq 3m_r$

Wenige Blätter pro Baum

Lemma
Die Reduktionsregeln 1 und 2 sorgen dafür, dass jeder Baum $\tau \in F$ maximal $c(\deg_1(\tau) + \deg_2(\tau))$ Blätter enthält (für kleine Konstante c).

Beweis

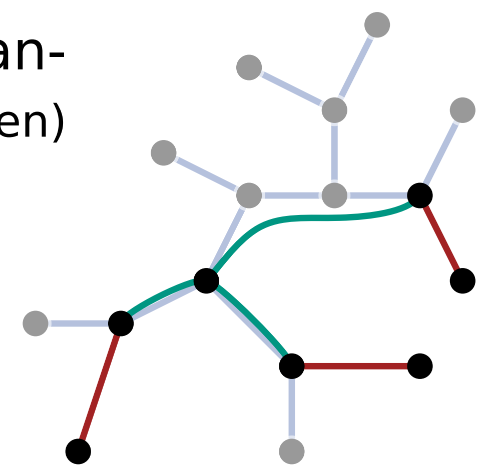
Zeige: n_1, n_2, m_g sind nicht viel größer als m_r

$(m_r \leq \deg_1(\tau) + \deg_2(\tau))$

n_d = #Grad-d Knoten in τ'
m_g = #grüne Kanten
m_r = #rote Kanten

- jedes Blatt hängt an roter Kante: $n_1 \leq m_r$ (Ausnahme: τ besteht aus einer Kante)
- jeder Knoten mit Grad 2 ist inzident zu einer roten Kante: $n_2 \leq 2m_r$ (sonst wäre er kontrahiert worden)
- Standardtrick in Bäumen – Kanten zählen:

$$m_r + m_g = n_1 + n_2 + n_3 - 1 \qquad \times 3$$
$$2(m_r + m_g) = n_1 + 2n_2 + 3n_3 \qquad -1$$
$$m_r + m_g = 2n_1 + n_2 - 3$$



- also: $m_g \leq 2n_1 + n_2 - m_r \leq 3m_r$

$\Rightarrow$ #Blätter in $\tau \leq n_1 + n_2 + 3m_g \leq 12m_r$

Zusammenfassung

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?

Theorem
Maximum Agreement Forest hat einen Kern mit $O(k)$ vielen Blättern, der in polynomieller Zeit berechnet werden kann.

Zusammenfassung

Problem: MAXIMUM AGREEMENT FOREST
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?

Theorem
MAXIMUM AGREEMENT FOREST hat einen Kern mit $O(k)$ vielen Blättern, der in polynomieller Zeit berechnet werden kann.

Methodik

- ein konkreteres Ziel als „ich will einen kleinen Kern“ kann helfen

(z.B.: ich will, dass jeder Baum in F nur wenige Blätter enthält)

Zusammenfassung

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?

Theorem
Maximum Agreement Forest hat einen Kern mit $O(k)$ vielen Blättern, der in polynomieller Zeit berechnet werden kann.

Methodik

- ein konkreteres Ziel als „ich will einen kleinen Kern“ kann helfen
  (z.B.: ich will, dass jeder Baum in F nur wenige Blätter enthält)
- Gegenbeispiele verraten, was wegreduziert werden muss

Zusammenfassung

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?

Theorem
Maximum Agreement Forest hat einen Kern mit $O(k)$ vielen Blättern, der in polynomieller Zeit berechnet werden kann.

Methodik

- ein konkreteres Ziel als „ich will einen kleinen Kern“ kann helfen
  (z.B.: ich will, dass jeder Baum in F nur wenige Blätter enthält)
- Gegenbeispiele verraten, was wegreduziert werden muss
- ggf. lohnt es, das Ziel später etwas aufzuweichen
  (z.B.: amortisiert über alle Bäume in F statt für jeden einzelnen)

Zusammenfassung

Karlsruher Institut für Technologie

Problem: Maximum Agreement Forest
Gegeben sind T_1, T_2 und ein Parameter k. Gibt es einen Agreement Forest mit maximal k Bäumen?

Theorem
Maximum Agreement Forest hat einen Kern mit $O(k)$ vielen Blättern, der in polynomieller Zeit berechnet werden kann.

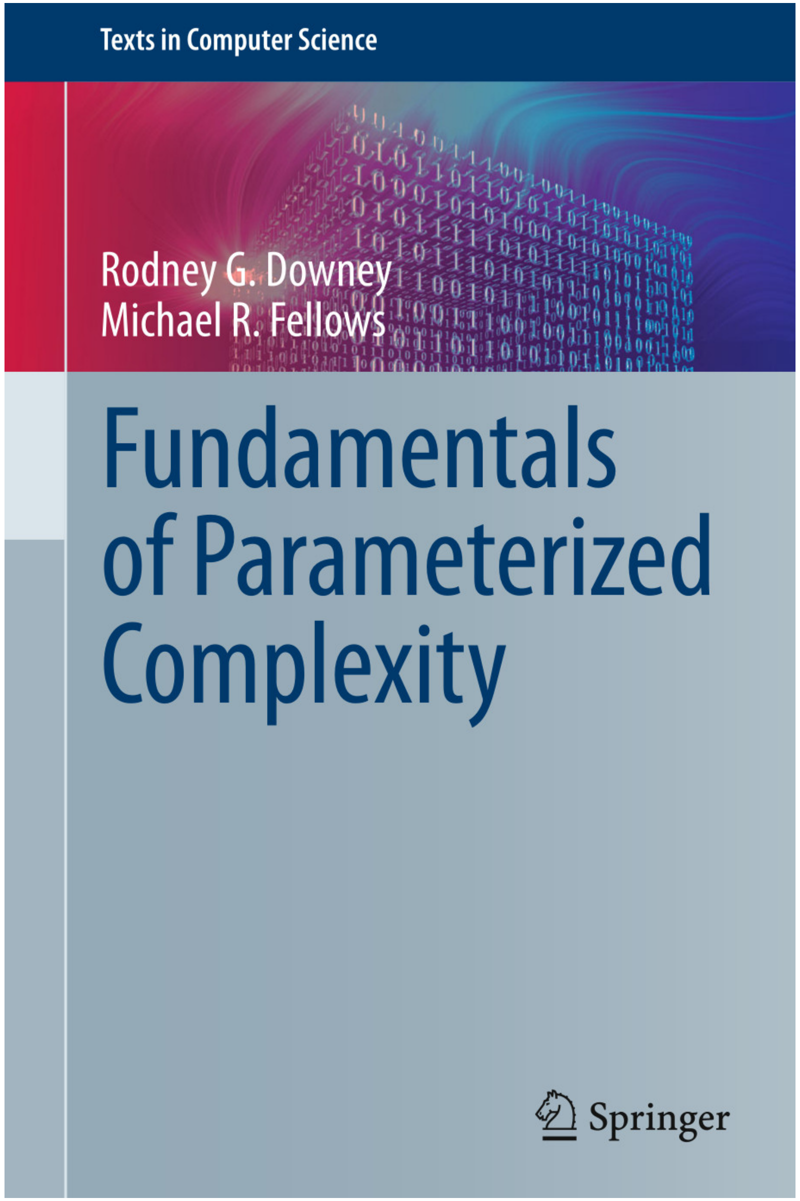
Methodik

- ein konkreteres Ziel als „ich will einen kleinen Kern“ kann helfen
  (z.B.: ich will, dass jeder Baum in F nur wenige Blätter enthält)
- Gegenbeispiele verraten, was wegreduziert werden muss
- ggf. lohnt es, das Ziel später etwas aufzuweichen
  (z.B.: amortisiert über alle Bäume in F statt für jeden einzelnen)

Nicht gesehen heute

- konkrete Laufzeit für Kernbildung
- konkrete Laufzeit für anschließendes Brute-Force im Kern
  $O(4^k \cdot k^5)$ ist möglich

Literaturhinweis



Anmerkungen

- Kapitel 4.10 handelt von dem eben betrachteten Thema
- enthält Links zur Originalliteratur
- aus dem Uninetz kostenlos abrufbar

`link.springer.com/book/10.1007/978-1-4471-5559-1`