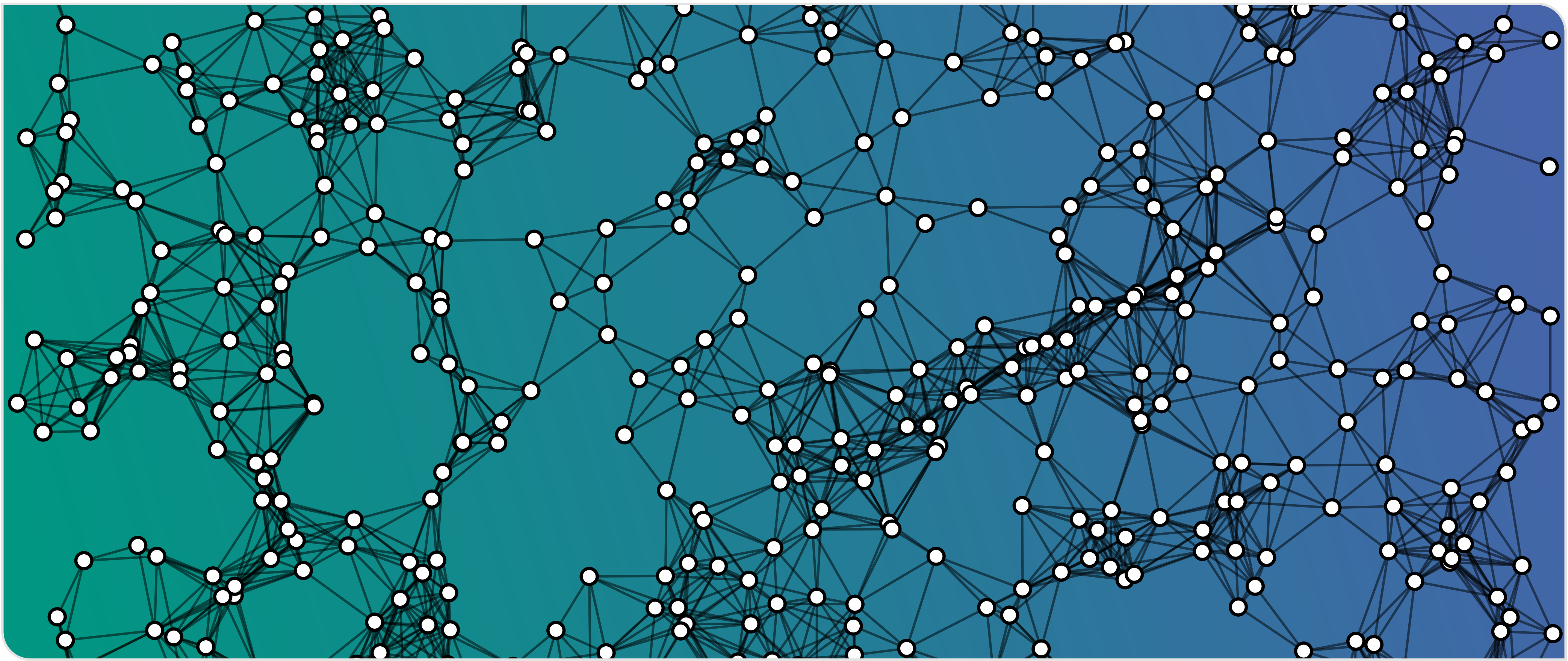
Karlsruher Institut für Technologie

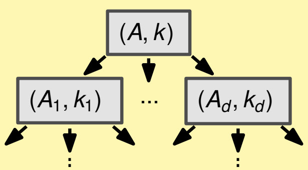
Parametrisierte Algorithmen

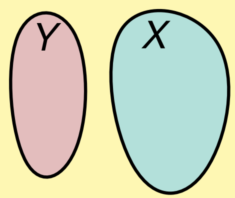
Iterative Kompression: Feedback Vertex Set

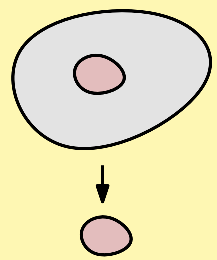


Inhalt

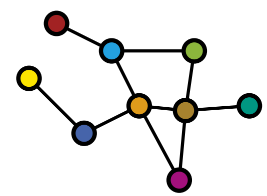
Basic Toolbox
- beschränkte Suchbäume
- iterative Kompression
- Kernbildung

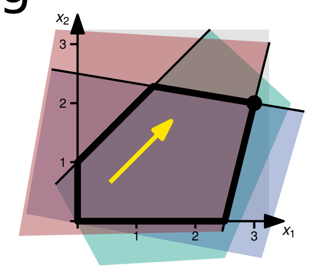


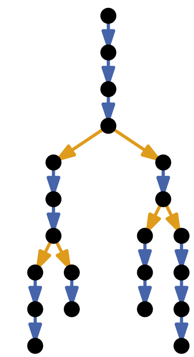




Erweiterte Toolbox
- lineare Programme
- Branch-and-Reduce
- Color Coding

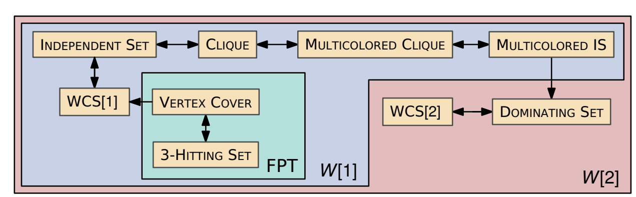


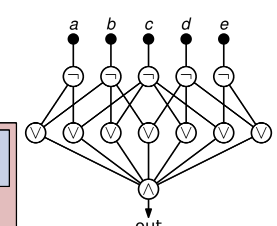




Baumweite
- dynamische Programme
- chordale & planare Graphen
- Courcelles Theorem

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



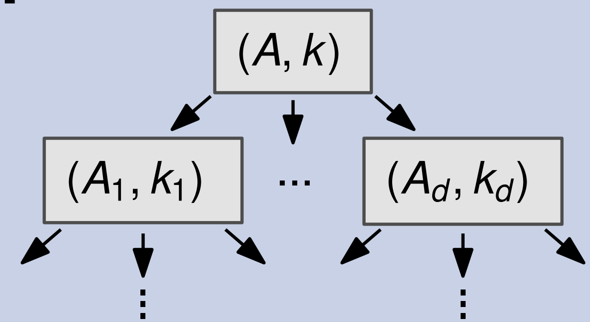


Wiederholung: Grundlegende Techniken

Beschränkter Suchbaum (Bounded Search Tree)

- für eine Instanz (A, k) bilde, in Zeit n^c, $(A_1, k_1), \dots, (A_d, k_d)$, sodass: (A, k) ist lösbar $\Leftrightarrow$ (A_i, k_i) ist lösbar für ein $i \in [1, d]$
- beschränke d durch eine Funktion $f_1(k)$
- beschränke Verzweigungstiefe durch $f_2(k)$

  (Beispiel: Parameter wird in jedem Schritt kleiner)
- $\Rightarrow$ FPT-Algo mit Laufzeit $O(f_1(k)^{f_2(k)} n^c)$



Kernbildung (Kernelization)

- wende sukzessive sichere Reduktionsregeln an
- zeige: übrig bleibt ein Kern, dessen Größe nur von k abhängt

Iterative Kompression (Iterative Compression)

- Kompressionsalgo: löst das Problem unter der Annahme eine etwas zu große Lösung zu kennen
- vergrößere Instanz schrittweise und halte initiale Lösung durch wiederholte Kompression klein

Feedback Vertex Set und Turniergraphen

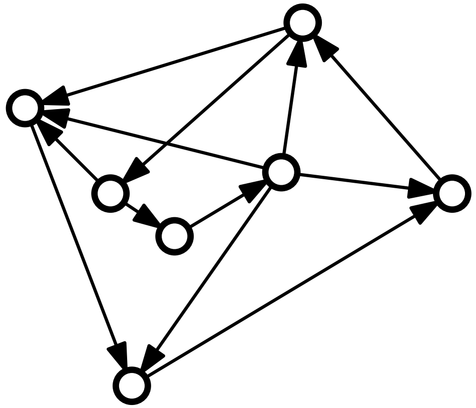
Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Feedback Vertex Set und Turniergraphen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?

($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel

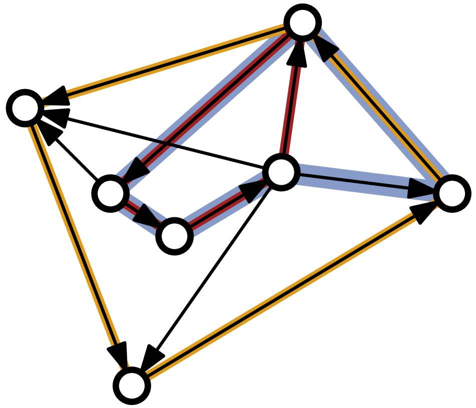


Feedback Vertex Set und Turniergraphen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?

($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel

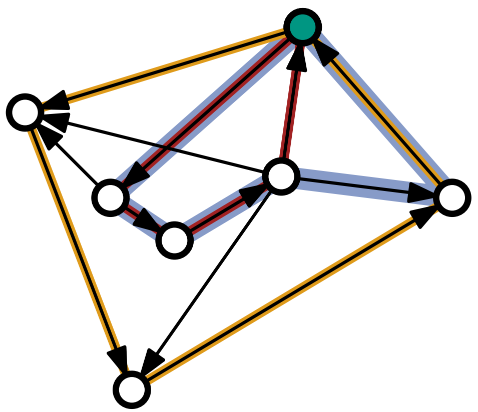


- nicht azyklisch

Feedback Vertex Set und Turniergraphen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



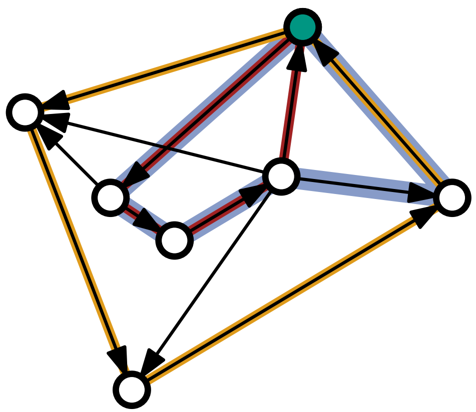
- nicht azyklisch
- alle gerichteten Kreise enthalten 

Feedback Vertex Set und Turniergraphen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?

($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel

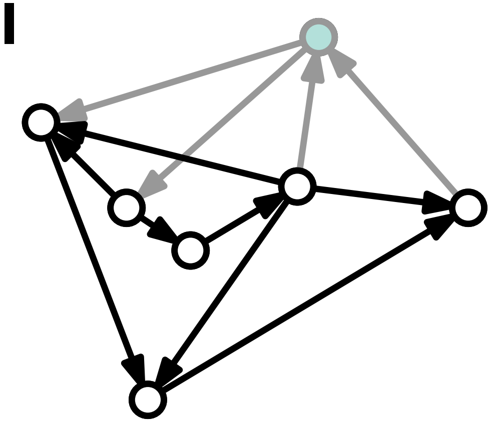


- nicht azyklisch
- alle gerichteten Kreise enthalten ●
- {●} ist Feedback Vertex Set der Größe 1

Feedback Vertex Set und Turniergraphen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel

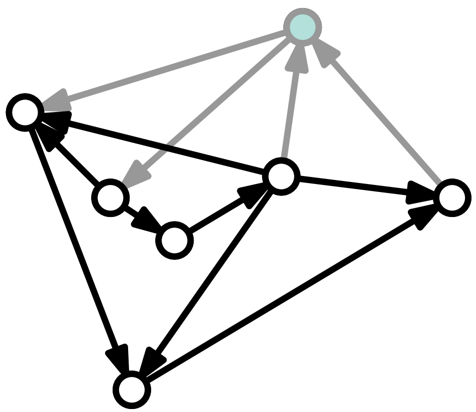


- nicht azyklisch
- alle gerichteten Kreise enthalten ●
- {●} ist Feedback Vertex Set der Größe 1

Feedback Vertex Set und Turniergraphen

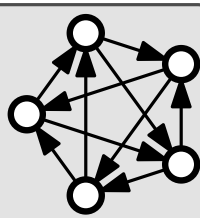
Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



- nicht azyklisch
- alle gerichteten Kreise enthalten ●
- { ● } ist Feedback Vertex Set der Größe 1

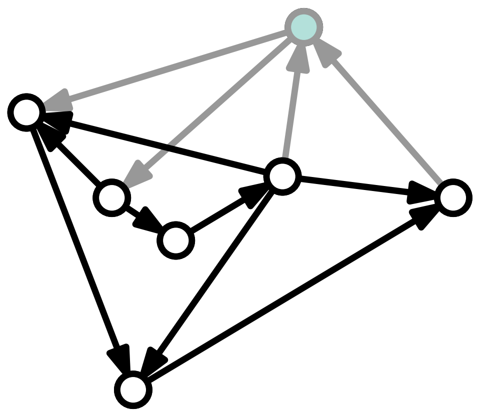
Definition
Ein $G = (V, E)$ ist ein **Turniergraph**, wenn für je zwei unterschiedliche Knoten $u, v \in V$ entweder $uv \in E$ oder $vu \in E$.



Feedback Vertex Set und Turniergraphen

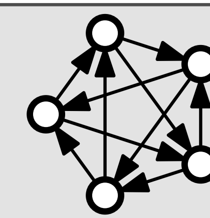
Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



- nicht azyklisch
- alle gerichteten Kreise enthalten
- $\{\ \}$ ist Feedback Vertex Set der Größe 1

Definition
Ein $G = (V, E)$ ist ein **Turniergraph**, wenn für je zwei unterschiedliche Knoten $u, v \in V$ entweder $uv \in E$ oder $vu \in E$.



Ziel

- FPT-Algorithmus für Feedback Vertex Set in Turniergraphen
- nutze dazu iterative Kompression

Kompressionsproblem

Problem: FEEDBACK VERTEX SET COMPRESSION
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Löse Feedback Vertex Set wie folgt

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Löse Feedback Vertex Set wie folgt

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Löse Feedback Vertex Set wie folgt

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i
  - falls $i \leq k$, dann ist V_i eine gültige Lösung

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Löse Feedback Vertex Set wie folgt

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i
  - falls $i \leq k$, dann ist V_i eine gültige Lösung
  - sonst, berechne rekursiv FVS F_{i-1} in G_{i-1}, sodass $|F_{i-1}| \leq k$

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Löse Feedback Vertex Set wie folgt

- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i
  - falls $i \leq k$, dann ist V_i eine gültige Lösung
  - sonst, berechne rekursiv FVS F_{i-1} in G_{i-1}, sodass $|F_{i-1}| \leq k$
  - $F_{i-1} \cup \{v_i\}$ ist FVS der Größe $k + 1$ für G_i

Kompressionsproblem

Problem: FEEDBACK VERTEX SET COMPRESSION
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Löse FEEDBACK VERTEX SET wie folgt

Problem: FEEDBACK VERTEX SET
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i
  - falls $i \leq k$, dann ist V_i eine gültige Lösung
  - sonst, berechne rekursiv FVS F_{i-1} in G_{i-1}, sodass $|F_{i-1}| \leq k$
  - $F_{i-1} \cup \{v_i\}$ ist FVS der Größe $k + 1$ für G_i
  - benutze Kompressionsalgo um FVS der Größe k zu bestimmen

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Löse Feedback Vertex Set wie folgt

- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i
  - falls $i \leq k$, dann ist V_i eine gültige Lösung
  - sonst, berechne rekursiv FVS F_{i-1} in G_{i-1}, sodass $|F_{i-1}| \leq k$
  - $F_{i-1} \cup \{v_i\}$ ist FVS der Größe $k + 1$ für G_i
  - benutze Kompressionsalgo um FVS der Größe k zu bestimmen

(gibt es keins, so können wir abbrechen, da es dann auch keins für G gibt)

Kompressionsproblem

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS der Größe $k + 1$. Gibt es ein FVS der Größe k?

Annahme

- wir können das Kompressionsproblem in $f(k) \cdot n^c$ Zeit lösen

Löse Feedback Vertex Set wie folgt

Problem: Feedback Vertex Set
Gegeben sei ein gerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

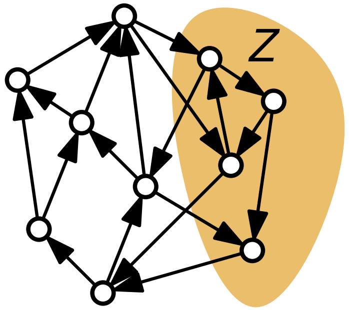
- sei $V_i = \{v_1, \dots, v_i\}$ und $G_i = G[V_i]$
- Ziel: berechne FVS der Größe maximal k für G_i
  - falls $i \leq k$, dann ist V_i eine gültige Lösung
  - sonst, berechne rekursiv FVS F_{i-1} in G_{i-1}, sodass $|F_{i-1}| \leq k$
  - $F_{i-1} \cup \{v_i\}$ ist FVS der Größe $k + 1$ für G_i
  - benutze Kompressionsalgo um FVS der Größe k zu bestimmen
    (gibt es keins, so können wir abbrechen, da es dann auch keins für G gibt)

Lemma
Wenn wir Feedback Vertex Set Compression in $f(k) \cdot n^c$ Zeit lösen können, dann können wir Feedback Vertex Set in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

Disjunkte Lösungen

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Z der Größe $k + 1$. Gibt es ein FVS der Größe k?

Idee

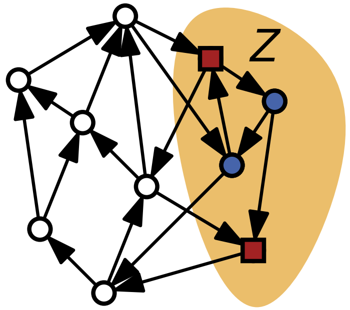


Disjunkte Lösungen

Problem: FEEDBACK VERTEX SET COMPRESSION
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Z der Größe $k + 1$. Gibt es ein FVS der Größe k?

Idee

- rate Teilmenge $X \subseteq Z$; $Y = Z \setminus X$
- gibt es FVS Z' mit $|Z'| \leq k$ und $Z' \cap Z = X$?

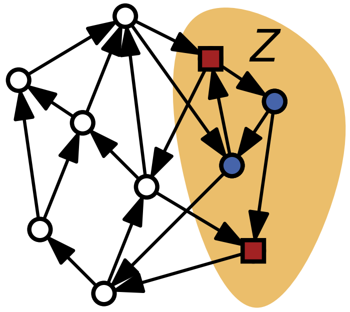


Disjunkte Lösungen

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Z der Größe $k + 1$. Gibt es ein FVS der Größe k?

Idee

- rate Teilmenge $X \subseteq Z$; $Y = Z \setminus X$
- gibt es FVS Z' mit $|Z'| \leq k$ und $Z' \cap Z = X$?
- äquivalent: gibt es FVS F in $G - X$ mit $|F| \leq k - |X|$ und $F \cap Y = \emptyset$?

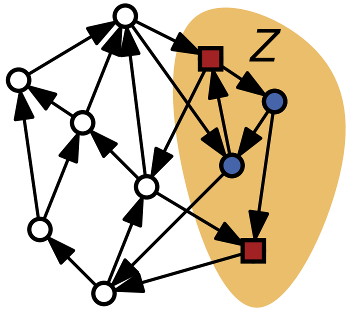


Disjunkte Lösungen

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Z der Größe $k + 1$. Gibt es ein FVS der Größe k?

Idee

- rate Teilmenge $X \subseteq Z$; $Y = Z \setminus X$
- gibt es FVS Z' mit $|Z'| \leq k$ und $Z' \cap Z = X$?
- äquivalent: gibt es FVS F in $G - X$ mit $|F| \leq k - |X|$ und $F \cap Y = \emptyset$?
- beachte: Y ist ein FVS der Größe $k - |X| + 1$ in $G - X$

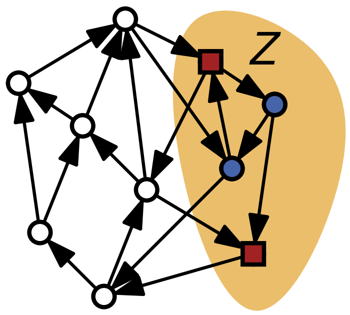


Disjunkte Lösungen

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Z der Größe $k + 1$. Gibt es ein FVS der Größe k?

Idee

- rate Teilmenge $X \subseteq Z$; $Y = Z \setminus X$
- gibt es FVS Z' mit $|Z'| \leq k$ und $Z' \cap Z = X$?
- äquivalent: gibt es FVS F in $G - X$ mit $|F| \leq k - |X|$ und $F \cap Y = \emptyset$?
- beachte: Y ist ein FVS der Größe $k - |X| + 1$ in $G - X$



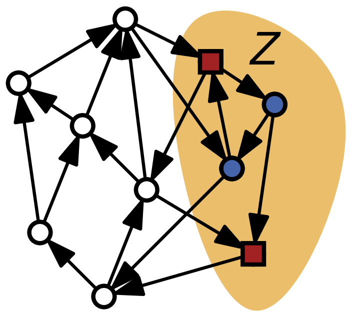
Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k + 1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Disjunkte Lösungen

Problem: Feedback Vertex Set Compression
Gegeben sei ein gerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Z der Größe $k + 1$. Gibt es ein FVS der Größe k?

Idee

- rate Teilmenge $X \subseteq Z$; $Y = Z \setminus X$
- gibt es FVS Z' mit $|Z'| \leq k$ und $Z' \cap Z = X$?
- äquivalent: gibt es FVS F in $G - X$ mit $|F| \leq k - |X|$ und $F \cap Y = \emptyset$?
- beachte: Y ist ein FVS der Größe $k - |X| + 1$ in $G - X$



Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k + 1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Einschub: spezielle Laufzeiten

Lemma
Wenn wir DISJOINT FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS COMPRESSION in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

DISJOINT-(PROBLEM) in polynomieller Zeit lösbar (n^c)

Einschub: spezielle Laufzeiten

Lemma
Wenn wir DISJOINT FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS COMPRESSION in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

DISJOINT-(PROBLEM) in polynomieller Zeit lösbar (n^c)

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} = 2^{k+1} - 1$
- resultierende Laufzeit: $O\left(\sum_{i=0}^{k} \binom{k+1}{i} \cdot n^c\right) = O(2^k \cdot n^c)$

Einschub: spezielle Laufzeiten

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Disjoint-(Problem) in polynomieller Zeit lösbar (n^c)

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} = 2^{k+1} - 1$
- resultierende Laufzeit: $O\left(\sum_{i=0}^{k} \binom{k+1}{i} \cdot n^c\right) = O(2^k \cdot n^c)$

Disjoint-(Problem) in $\alpha^k \cdot n^c$ lösbar

Einschub: spezielle Laufzeiten

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Disjoint-(Problem) in polynomieller Zeit lösbar (n^c)

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} = 2^{k+1} - 1$
- resultierende Laufzeit: $O\left(\sum_{i=0}^{k} \binom{k+1}{i} \cdot n^c\right) = O(2^k \cdot n^c)$

Disjoint-(Problem) in $\alpha^k \cdot n^c$ lösbar

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} \alpha^{k-i} \leq (k+1) \cdot \sum_{i=0}^{k} \binom{k}{i} \alpha^{k-i}$

Einschub: spezielle Laufzeiten

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Disjoint-(Problem) in polynomieller Zeit lösbar (n^c)

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} = 2^{k+1} - 1$
- resultierende Laufzeit: $O\left(\sum_{i=0}^{k} \binom{k+1}{i} \cdot n^c\right) = O(2^k \cdot n^c)$

Disjoint-(Problem) in $\alpha^k \cdot n^c$ lösbar

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} \alpha^{k-i} \leq (k+1) \cdot \sum_{i=0}^{k} \binom{k}{i} \alpha^{k-i}$

Erinnerung: $(a+b)^k =$

Einschub: spezielle Laufzeiten

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Disjoint-(Problem) in polynomieller Zeit lösbar (n^c)

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} = 2^{k+1} - 1$
- resultierende Laufzeit: $O\left(\sum_{i=0}^{k} \binom{k+1}{i} \cdot n^c\right) = O(2^k \cdot n^c)$

Disjoint-(Problem) in $\alpha^k \cdot n^c$ lösbar

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} \alpha^{k-i} \leq (k+1) \cdot \sum_{i=0}^{k} \binom{k}{i} \alpha^{k-i}$

Erinnerung: $(a+b)^k = \sum_{i=0}^{k} \binom{k}{i} a^i b^{k-i}$

Einschub: spezielle Laufzeiten

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Disjoint-(Problem) in polynomieller Zeit lösbar (n^c)

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} = 2^{k+1} - 1$
- resultierende Laufzeit: $O\left(\sum_{i=0}^{k} \binom{k+1}{i} \cdot n^c\right) = O(2^k \cdot n^c)$

Disjoint-(Problem) in $\alpha^k \cdot n^c$ lösbar

- es gilt: $\sum_{i=0}^{k} \binom{k+1}{i} \alpha^{k-i} \leq (k+1) \cdot \sum_{i=0}^{k} \binom{k}{i} \alpha^{k-i}$
- resultierende Laufzeit: $O(k(1+\alpha)^k \cdot n^c)$

Erinnerung: $(a+b)^k = \sum_{i=0}^{k} \binom{k}{i} a^i b^{k-i}$

Aktueller Stand

Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k + 1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Lemma
Wenn wir Feedback Vertex Set Compression in $f(k) \cdot n^c$ Zeit lösen können, dann können wir Feedback Vertex Set in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

Aktueller Stand

Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Lemma
Wenn wir Feedback Vertex Set Compression in $f(k) \cdot n^c$ Zeit lösen können, dann können wir Feedback Vertex Set in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

Ziel
- FPT-Algorithmus für Feedback Vertex Set in Turniergraphen

Aktueller Stand

Problem: DISJOINT FEEDBACK VERTEX SET
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Lemma
Wenn wir DISJOINT FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS COMPRESSION in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Lemma
Wenn wir FEEDBACK VERTEX SET COMPRESSION in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FEEDBACK VERTEX SET in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

Ziel
- FPT-Algorithmus für FEEDBACK VERTEX SET in Turniergraphen
- ausreichend: FPT-Algorithmus für DISJOINT FVS in Turniergraphen

Aktueller Stand

Problem: DISJOINT FEEDBACK VERTEX SET
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Lemma
Wenn wir DISJOINT FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS COMPRESSION in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Lemma
Wenn wir FEEDBACK VERTEX SET COMPRESSION in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FEEDBACK VERTEX SET in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

Ziel

- FPT-Algorithmus für FEEDBACK VERTEX SET in Turniergraphen
- ausreichend: FPT-Algorithmus für DISJOINT FVS in Turniergraphen

Beachte

- bisher nicht genutzt, dass G ein Turniergraph ist

Aktueller Stand

Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Lemma
Wenn wir Feedback Vertex Set Compression in $f(k) \cdot n^c$ Zeit lösen können, dann können wir Feedback Vertex Set in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

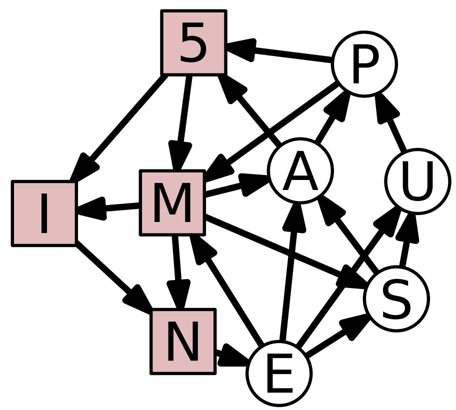
Ziel
- FPT-Algorithmus für Feedback Vertex Set in Turniergraphen
- ausreichend: FPT-Algorithmus für Disjoint FVS in Turniergraphen

Beachte
- bisher nicht genutzt, dass G ein Turniergraph ist
- jeder induzierte Teilgraph eines Turniergraphen ist ein Turniergraph

Finde ein minimales FVS ohne Y

Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k + 1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

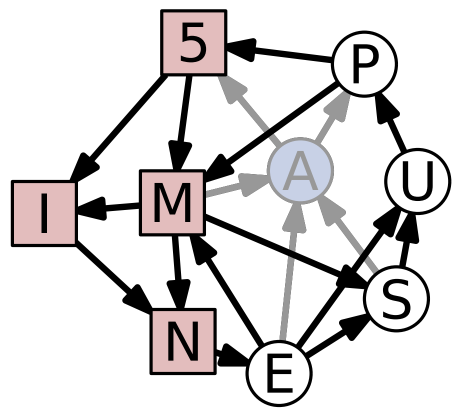
$Y = \{\boxed{5}, \boxed{I}, \boxed{M}, \boxed{N}\}$



Finde ein minimales FVS ohne Y

Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

$Y = \{\boxed{5}, \boxed{\mathrm{I}}, \boxed{\mathrm{M}}, \boxed{\mathrm{N}}\}$



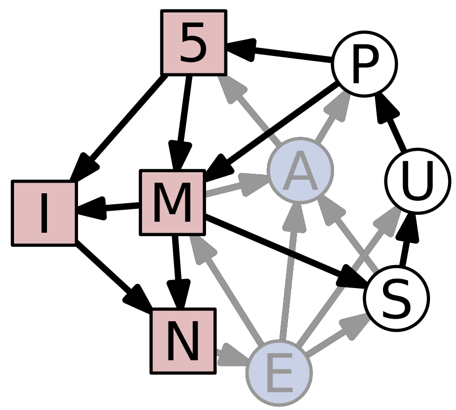
Lösung

- A muss gewählt werden (wegen dem Dreieck $5MA$)

Finde ein minimales FVS ohne Y

Problem: DISJOINT FEEDBACK VERTEX SET
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

$Y = \{\boxed{5}, \boxed{I}, \boxed{M}, \boxed{N}\}$



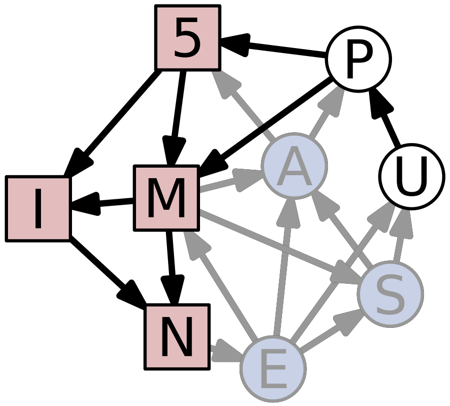
Lösung

- A muss gewählt werden (wegen dem Dreieck $5MA$)
- E muss gewählt werden (wegen dem Dreieck MNE)

Finde ein minimales FVS ohne Y

Problem: Disjoint Feedback Vertex Set
Gegeben sei ein gerichteter Graph G, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

$Y = \{\boxed{5}, \boxed{I}, \boxed{M}, \boxed{N}\}$



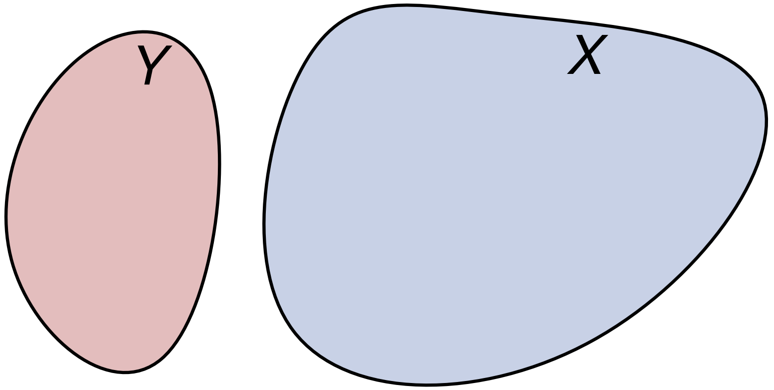
Lösung

- A muss gewählt werden (wegen dem Dreieck $5MA$)
- E muss gewählt werden (wegen dem Dreieck MNE)
- außerdem muss noch P, U oder S gewählt werden

DISJOINT FVS in Turniergraphen

Situation

- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch



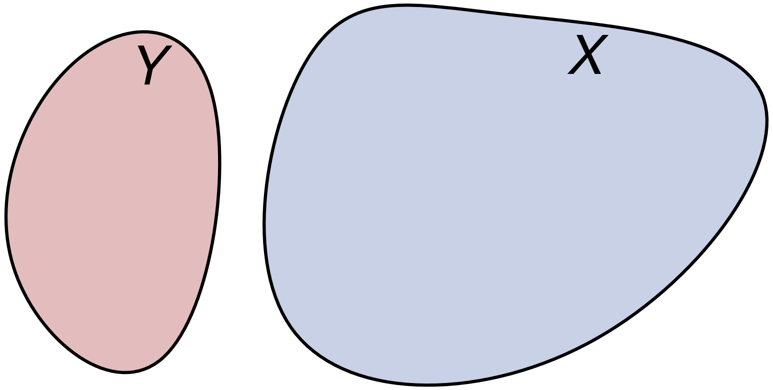
Disjoint FVS in Turniergraphen

Situation

- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch

Beobachtung

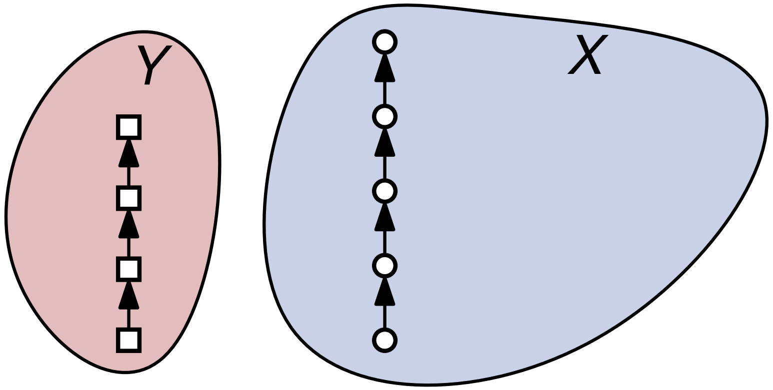
- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch



DISJOINT FVS in Turniergraphen

Situation

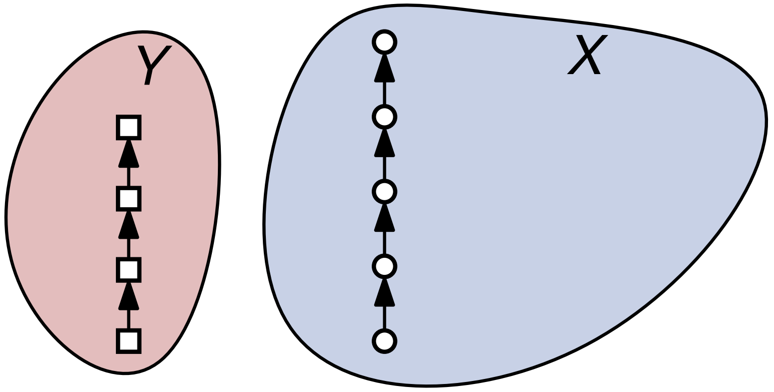
- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch



Beobachtung

- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

DISJOINT FVS in Turniergraphen



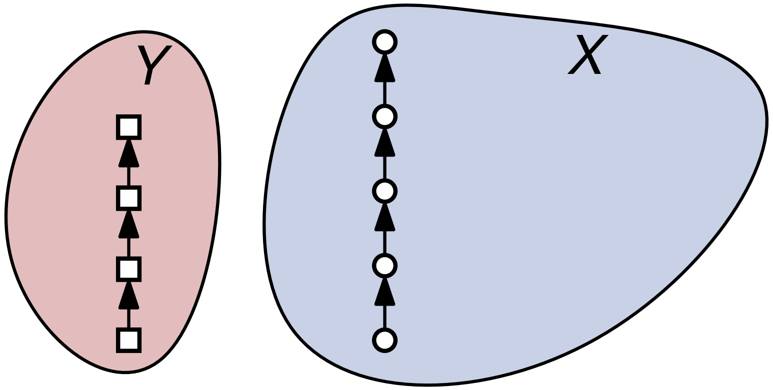
Situation
- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch

Beobachtung
- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow$ $G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

Idee
- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein

Disjoint FVS in Turniergraphen

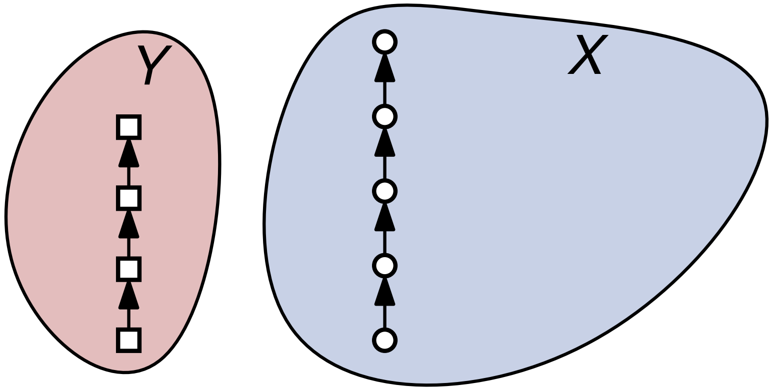
Situation
- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch



Beobachtung
- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

Idee
- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein
- daraus resultierende Ordnung auf X darf der durch $G[X]$ gegebenen Ordnung nicht widersprechen

Disjoint FVS in Turniergraphen

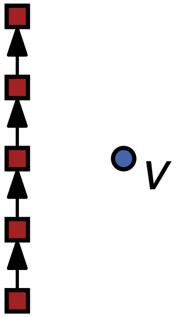


Situation
- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch

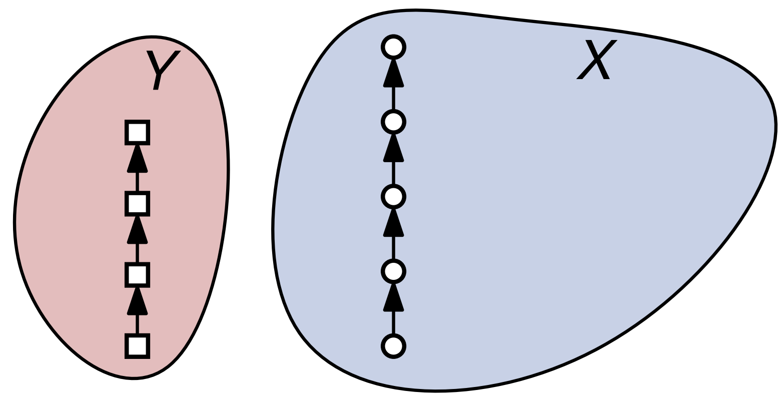
Beobachtung
- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

Idee
- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein
- daraus resultierende Ordnung auf X darf der durch $G[X]$ gegebenen Ordnung nicht widersprechen

Einsortieren eines einzelnen Knotens



DISJOINT FVS in Turniergraphen



Situation

- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch

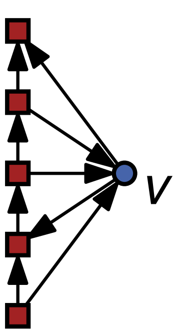
Beobachtung

- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

Idee

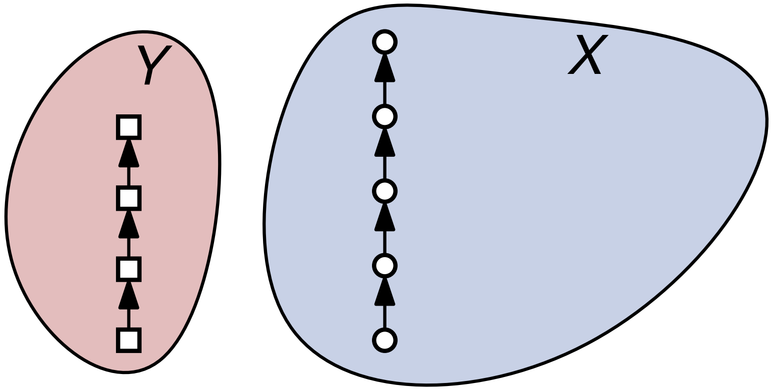
- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein
- daraus resultierende Ordnung auf X darf der durch $G[X]$ gegebenen Ordnung nicht widersprechen

Einsortieren eines einzelnen Knotens



- wenn $G[Y] + v$ zyklisch $\Rightarrow$ lösche v und verringere k um 1

Disjoint FVS in Turniergraphen



Situation

- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch

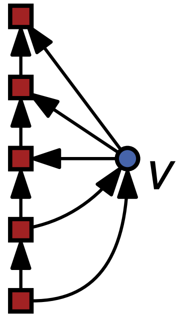
Beobachtung

- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

Idee

- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein
- daraus resultierende Ordnung auf X darf der durch $G[X]$ gegebenen Ordnung nicht widersprechen

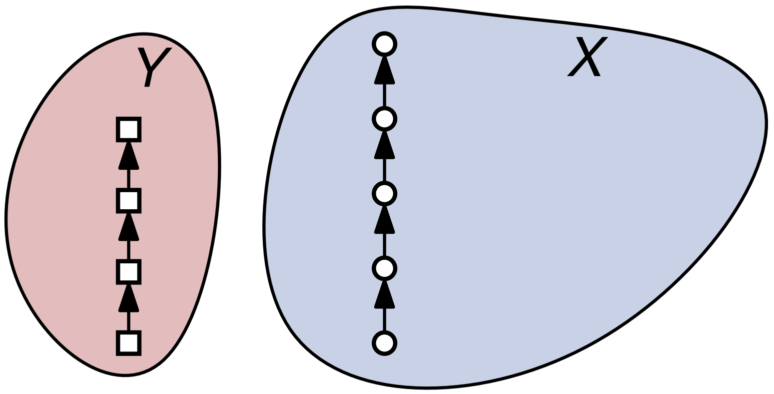
Einsortieren eines einzelnen Knotens



- wenn $G[Y] + v$ zyklisch $\Rightarrow$ lösche v und verringere k um 1
- sonst: eingehende/ausgehende Kanten sind konsekutiv

Disjoint FVS in Turniergraphen

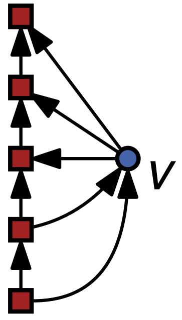
Situation
- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch



Beobachtung
- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

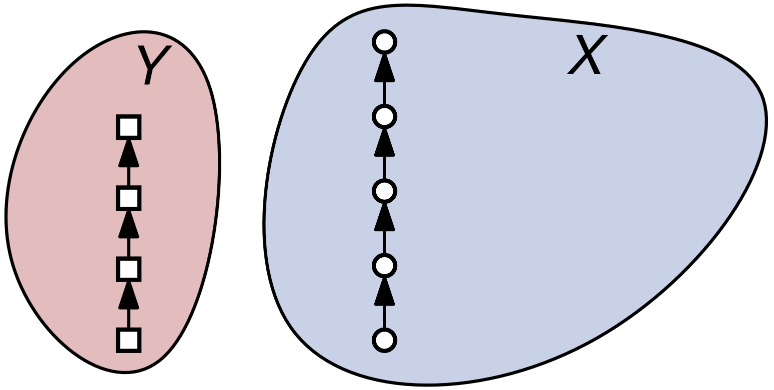
Idee
- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein
- daraus resultierende Ordnung auf X darf der durch $G[X]$ gegebenen Ordnung nicht widersprechen

Einsortieren eines einzelnen Knotens



- wenn $G[Y] + v$ zyklisch $\Rightarrow$ lösche v und verringere k um 1
- sonst: eingehende/ausgehende Kanten sind konsekutiv
- Position von v eindeutig festgelegt

Disjoint FVS in Turniergraphen



Situation

- wähle Knoten aus X und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch

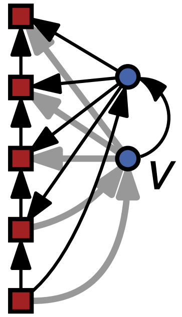
Beobachtung

- X muss ebenfalls ein FVS sein (sonst: nein-Instanz) $\Rightarrow$ $G[Y]$ ist azyklisch
- azyklische Turniergraphen definieren totale Ordnungen

Idee

- sortiere möglichst viele Knoten aus X in die Ordnung auf Y ein
- daraus resultierende Ordnung auf X darf der durch $G[X]$ gegebenen Ordnung nicht widersprechen

Einsortieren eines einzelnen Knotens

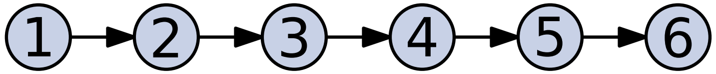


- wenn $G[Y] + v$ zyklisch $\Rightarrow$ lösche v und verringere k um 1
- sonst: eingehende/ausgehende Kanten sind konsekutiv
- Position von v eindeutig festgelegt
- Konflikt, wenn Nachfolger von v vor v einsortiert wird

Zwei Ordnungen

Beispiel

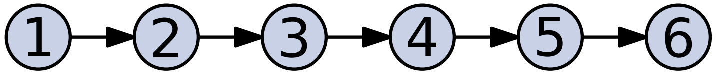
- Ordnung auf X gegeben durch $G[X]$

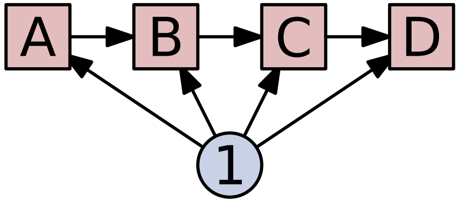


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

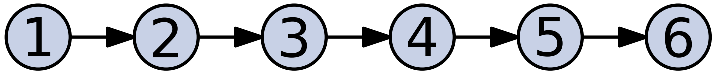


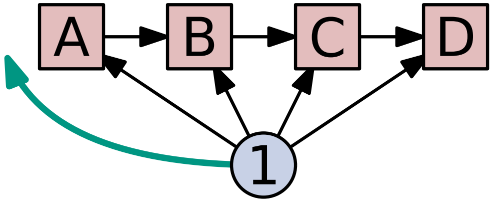


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

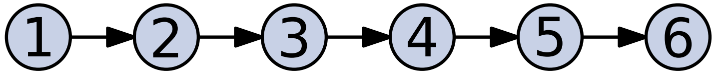


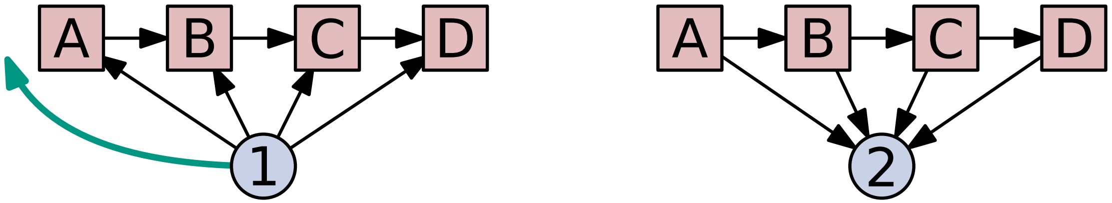


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

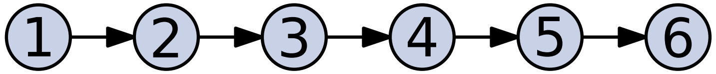


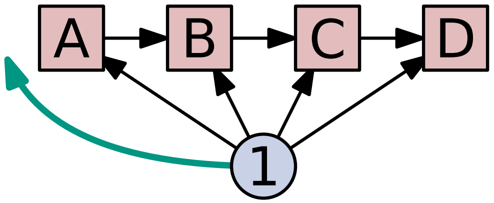


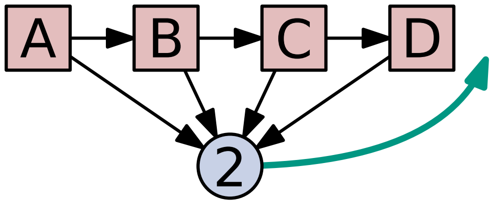
Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$



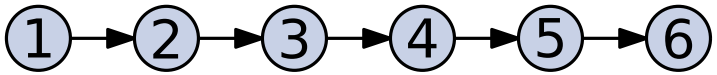


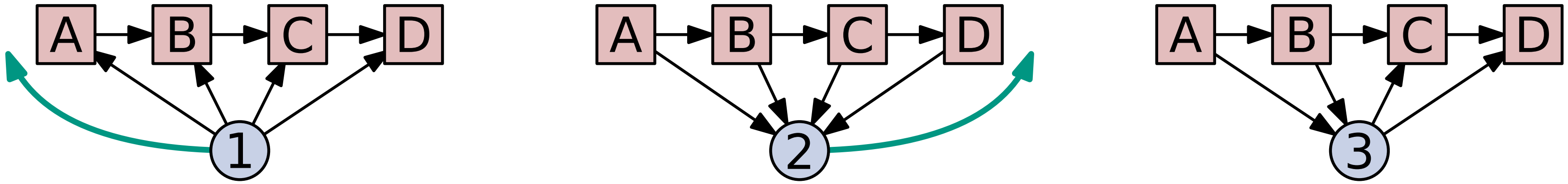


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

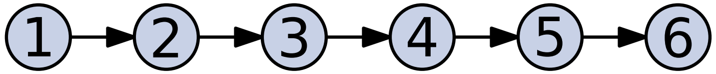


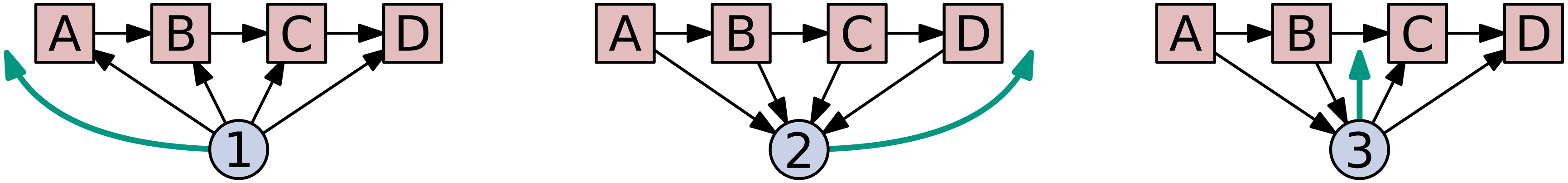


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

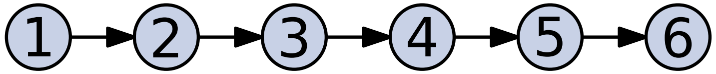


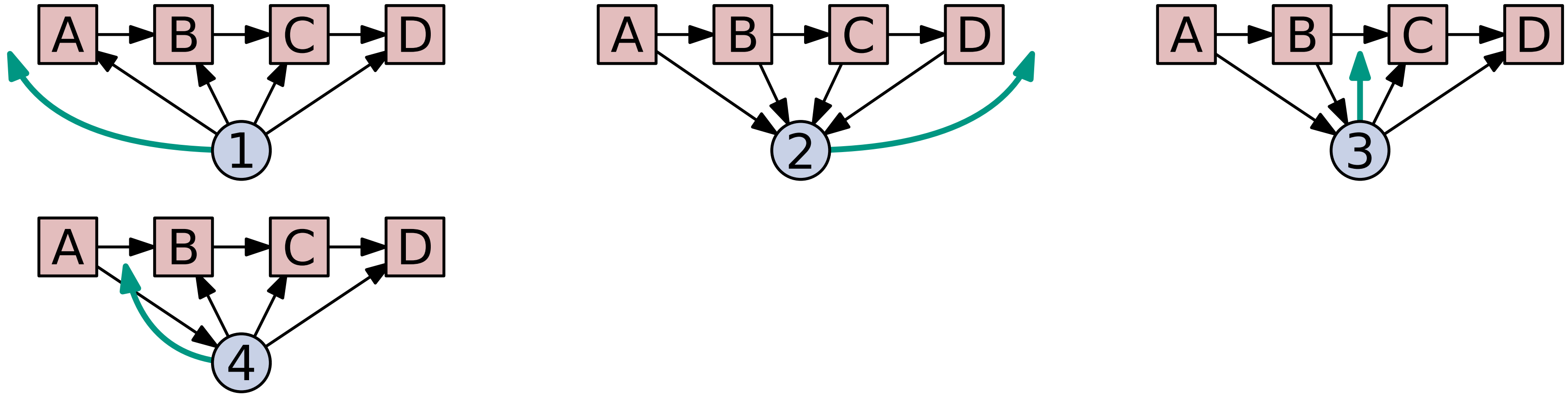


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

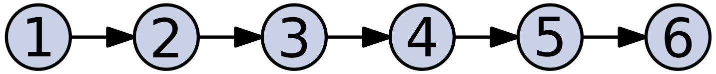


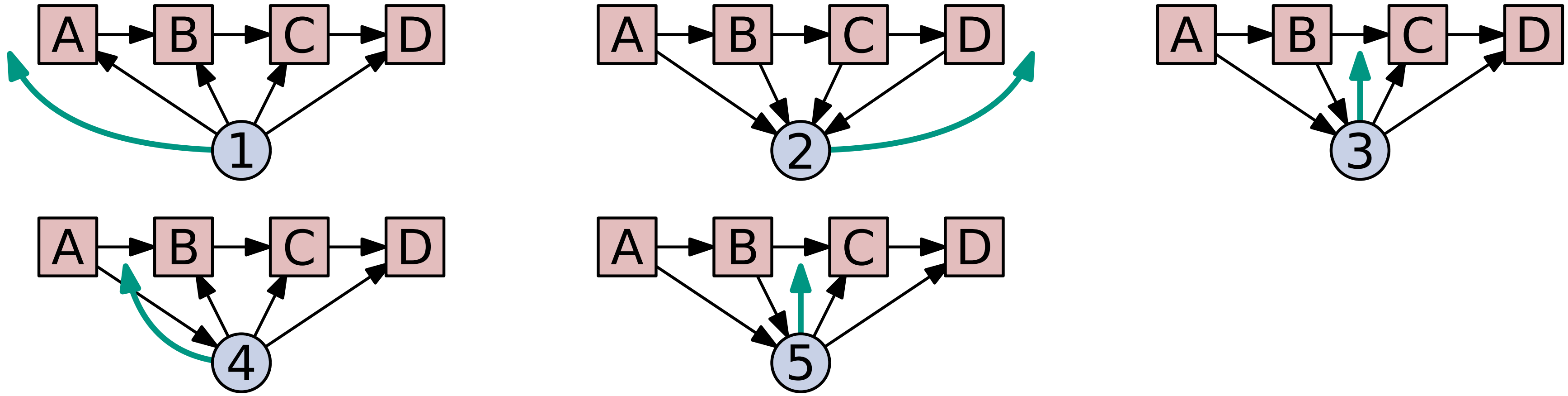


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

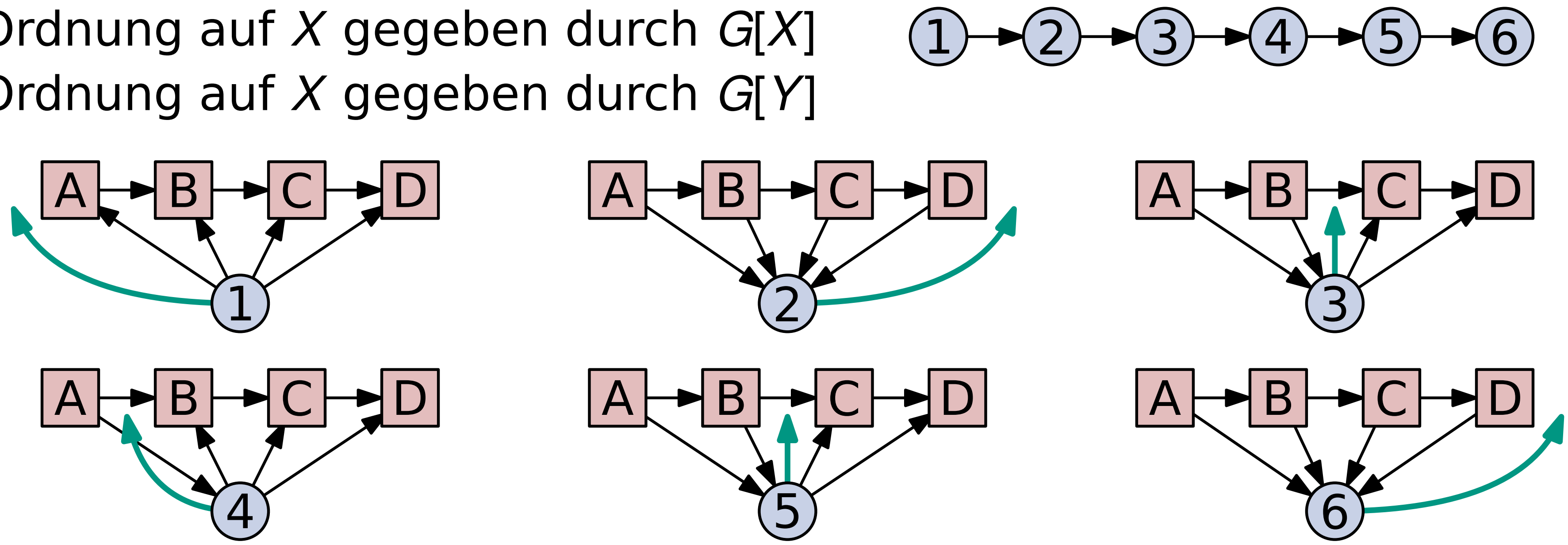




Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

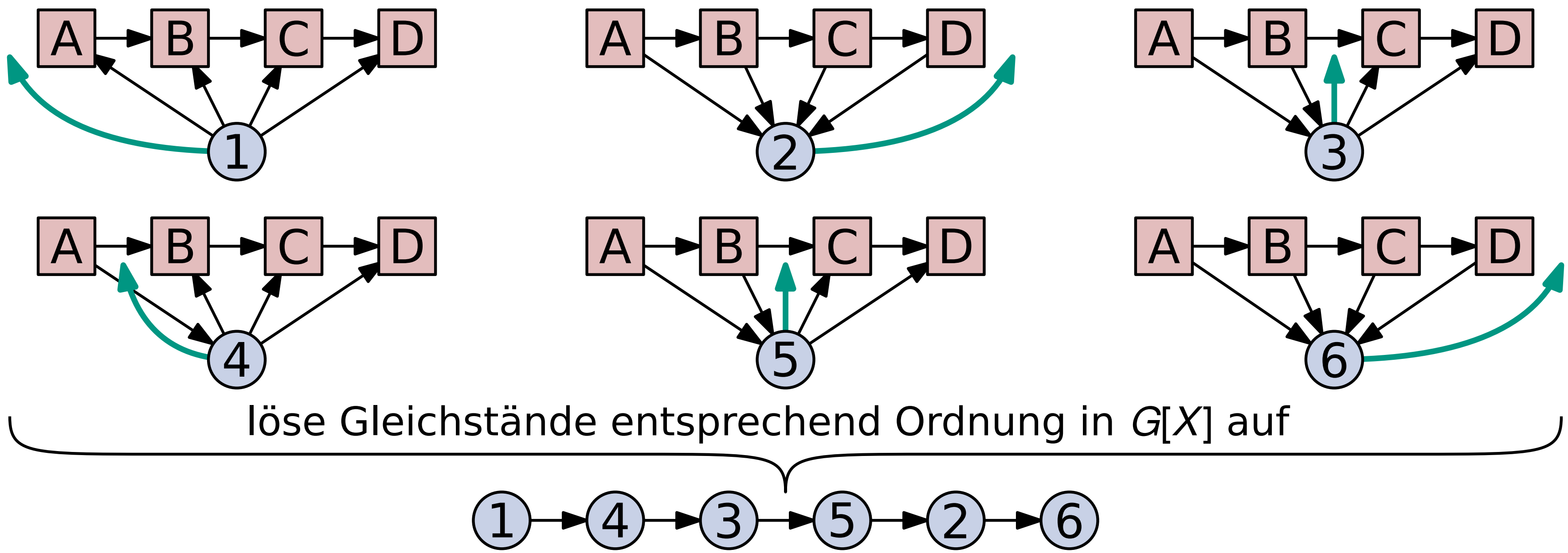


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$

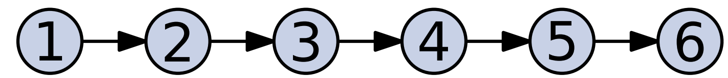
$1 \to 2 \to 3 \to 4 \to 5 \to 6$

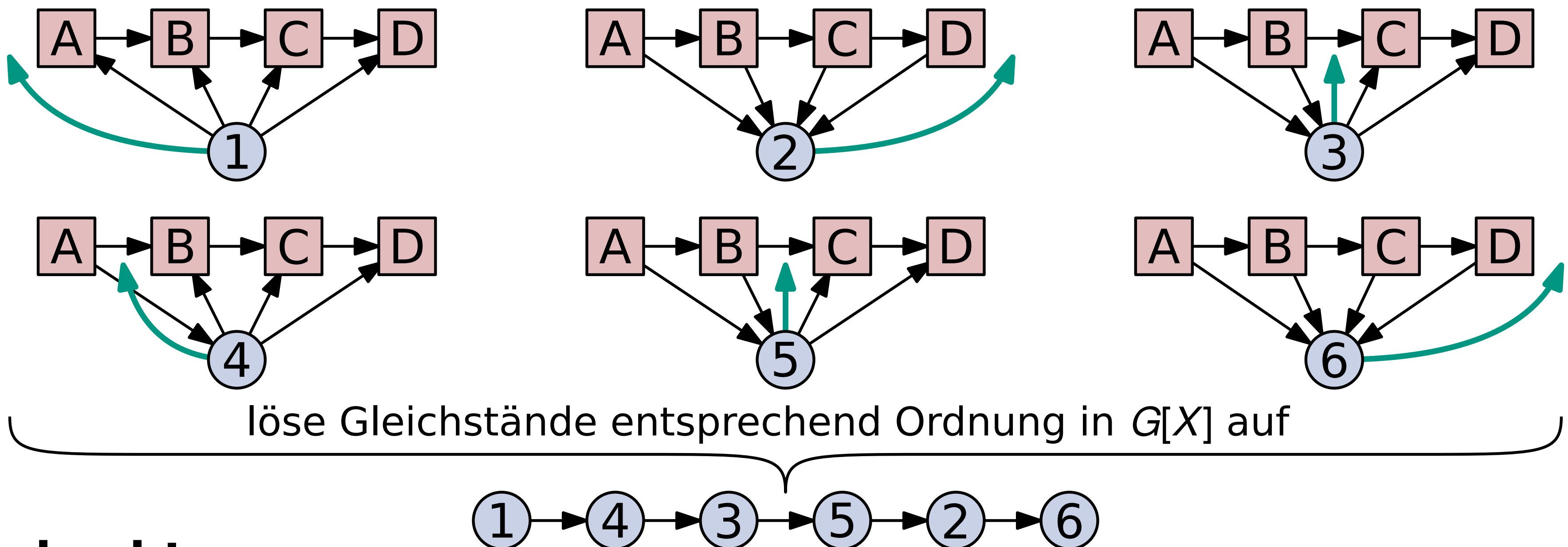


Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$





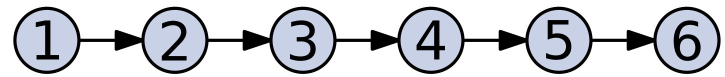
Beobachtung

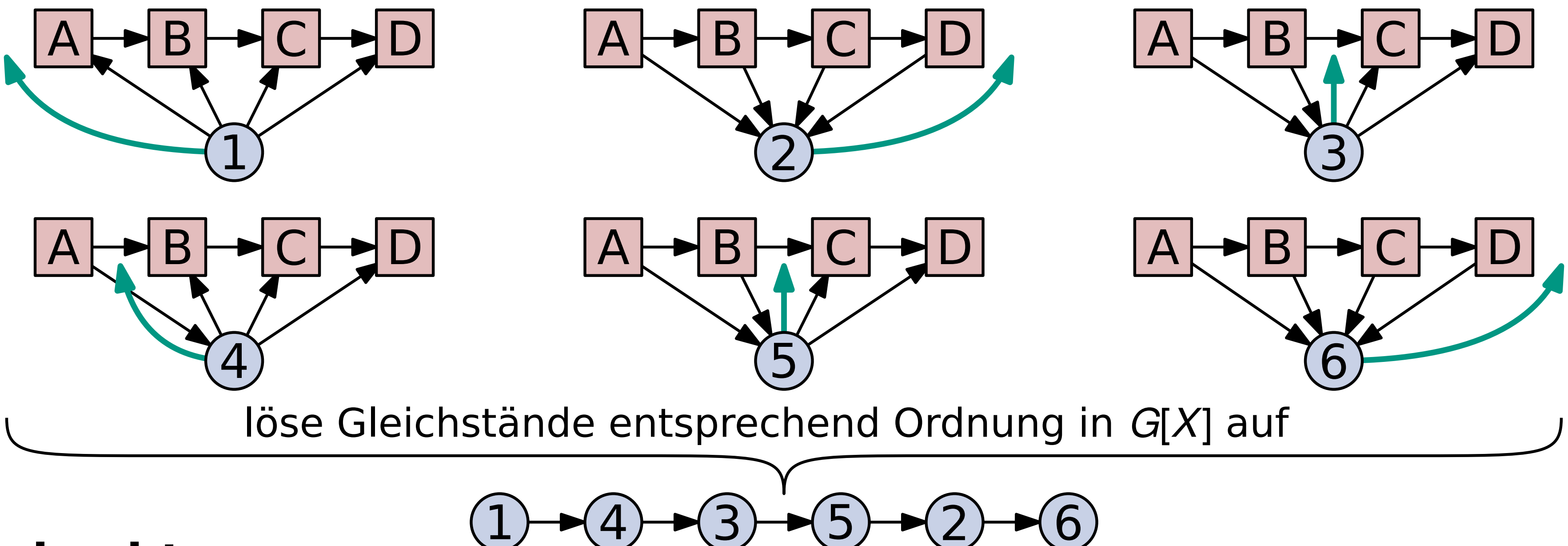
- F ist ein FVS $\Leftrightarrow$ beide Ordnungen sind gleich auf $X \setminus F$

Zwei Ordnungen

Beispiel

- Ordnung auf X gegeben durch $G[X]$
- Ordnung auf X gegeben durch $G[Y]$





Beobachtung

- F ist ein FVS $\Leftrightarrow$ beide Ordnungen sind gleich auf $X \setminus F$

Ziel

- finde möglichst große Teilmenge von X, für die beide Ordnungen übereinstimmen $\rightarrow$ Longest Common Subsequence

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \ldots, a_p$ und $b_1, \ldots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$
  - finde längste gem. Teilfolge von $a_1, \dots, a_{p-1}$ und $b_1, \dots, b_{q-1}$
  - hänge $a_p = b_q$ an diese Teilfolge an

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$
  - finde längste gem. Teilfolge von $a_1, \dots, a_{p-1}$ und $b_1, \dots, b_{q-1}$
  - hänge $a_p = b_q$ an diese Teilfolge an
- falls $a_p \neq b_q$

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$
  - finde längste gem. Teilfolge von $a_1, \dots, a_{p-1}$ und $b_1, \dots, b_{q-1}$
  - hänge $a_p = b_q$ an diese Teilfolge an
- falls $a_p \neq b_q$
  - lösche entweder a_p oder b_q

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$
  - finde längste gem. Teilfolge von $a_1, \dots, a_{p-1}$ und $b_1, \dots, b_{q-1}$
  - hänge $a_p = b_q$ an diese Teilfolge an
- falls $a_p \neq b_q$
  - lösche entweder a_p oder b_q
  - berechne längste gem. Teilfolgen von $a_1, \dots, a_p$ und $b_1, \dots b_{q-1}$ sowie von $a_1, \dots, a_{p-1}$ und $b_1, \dots b_q$ $\rightarrow$ wähle Maximum

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$
  - finde längste gem. Teilfolge von $a_1, \dots, a_{p-1}$ und $b_1, \dots, b_{q-1}$
  - hänge $a_p = b_q$ an diese Teilfolge an
- falls $a_p \neq b_q$
  - lösche entweder a_p oder b_q
  - berechne längste gem. Teilfolgen von $a_1, \dots, a_p$ und $b_1, \dots b_{q-1}$ sowie von $a_1, \dots, a_{p-1}$ und $b_1, \dots b_q$ $\to$ wähle Maximum
- das liefert ein dynamisches Programm mit Laufzeit $O(pq)$

Longest Common Subsequence

Problem: Longest Common Subsequence
Gegeben seien zwei Folgen $a_1, \dots, a_p$ und $b_1, \dots, b_q$. Gesucht ist die längste Folge, die Teilfolge beider Folgen ist.

Beobachtung

- falls $a_p = b_q$
  - finde längste gem. Teilfolge von $a_1, \dots, a_{p-1}$ und $b_1, \dots, b_{q-1}$
  - hänge $a_p = b_q$ an diese Teilfolge an
- falls $a_p \neq b_q$
  - lösche entweder a_p oder b_q
  - berechne längste gem. Teilfolgen von $a_1, \dots, a_p$ und $b_1, \dots b_{q-1}$ sowie von $a_1, \dots, a_{p-1}$ und $b_1, \dots b_q \to$ wähle Maximum
- das liefert ein dynamisches Programm mit Laufzeit $O(pq)$

Lemma
Longest Common Subsequence für zwei Folgen der Länge p bzw. q kann in $O(pq)$ Zeit gelöst werden.

Feedback Vertex Set auf Turniergraphen

Disjoint Feedback Vertex Set: $O(n^2)$ (via Longest Common Subsequence)

Lemma
Wenn wir Disjoint FVS in $f(k) \cdot n^c$ Zeit lösen können, dann können wir FVS Compression in $O\left(\sum_{i=0}^{k} \binom{k+1}{i} f(k-i) \cdot n^c\right)$ Zeit lösen.

Feedback Vertex Set Compression: $O(2^k \cdot n^2)$

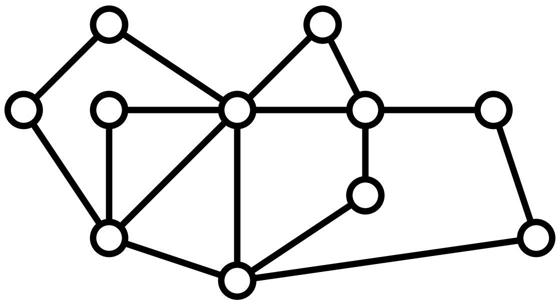
Lemma
Wenn wir Feedback Vertex Set Compression in $f(k) \cdot n^c$ Zeit lösen können, dann können wir Feedback Vertex Set in $O(f(k) \cdot n^{c+1})$ Zeit lösen.

Theorem
Feedback Vertex Set kann auf Turniergraphen in $O(2^k \cdot n^3)$ Zeit gelöst werden.

Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k.
Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

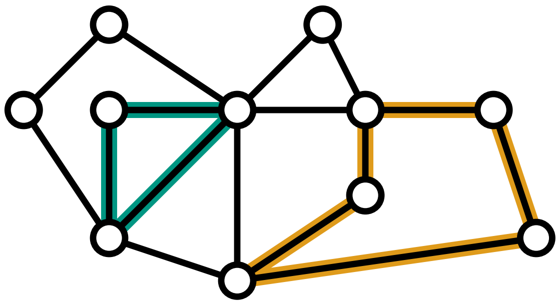
Beispiel



Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k.
Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel

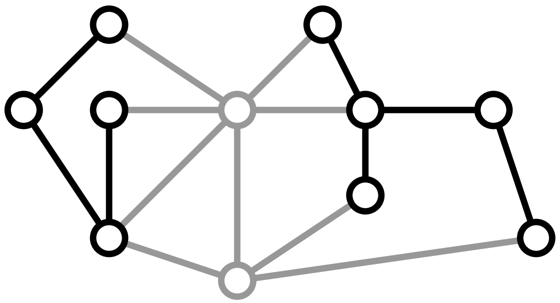


- wir müssen alle Kreise erwischen
- mindestens zwei Knoten muss man löschen

Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k.
Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel

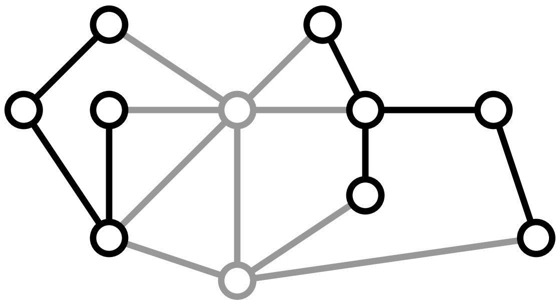


- wir müssen alle Kreise erwischen
- mindestens zwei Knoten muss man löschen
- zwei reichen auch aus

Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k.
Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



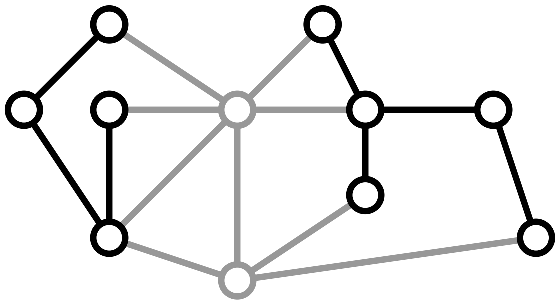
- wir müssen alle Kreise erwischen
- mindestens zwei Knoten muss man löschen
- zwei reichen auch aus

Ziel: FPT-Algorithmus für Feedback Vertex Set

Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k.
Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



- wir müssen alle Kreise erwischen
- mindestens zwei Knoten muss man löschen
- zwei reichen auch aus

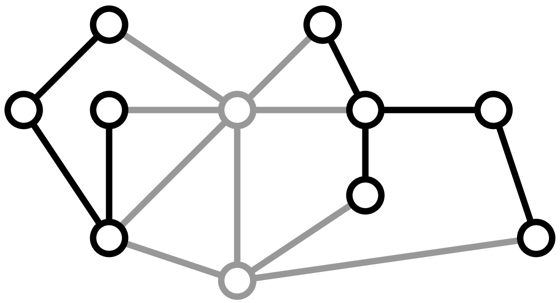
Ziel: FPT-Algorithmus für Feedback Vertex Set

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k. Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



- wir müssen alle Kreise erwischen
- mindestens zwei Knoten muss man löschen
- zwei reichen auch aus

Ziel: FPT-Algorithmus für Feedback Vertex Set

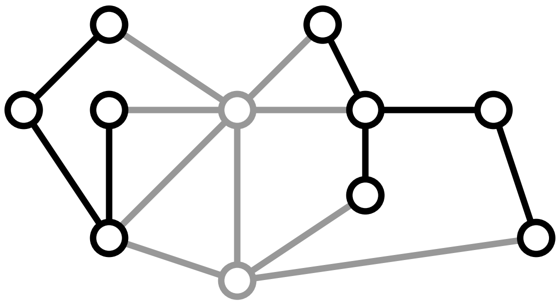
Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Liefert ein FPT-Algo für Disjoint FVS auch hier wieder einen FPT-Algo für FVS?

Nochmal Feedback Vertex Set

Problem: Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$ und ein Parameter k.
Gibt es ein Feedback Vertex Set der Größe k?
($F \subseteq V$ ist ein *Feedback Vertex Set*, wenn $G - F$ azyklisch ist)

Beispiel



- wir müssen alle Kreise erwischen
- mindestens zwei Knoten muss man löschen
- zwei reichen auch aus

Ziel: FPT-Algorithmus für Feedback Vertex Set

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Liefert ein FPT-Algo für Disjoint FVS auch hier wieder einen FPT-Algo für FVS?

Ja!

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

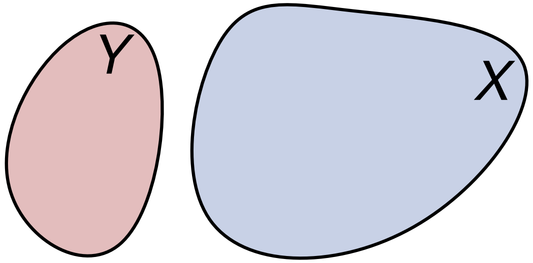
Situation

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y

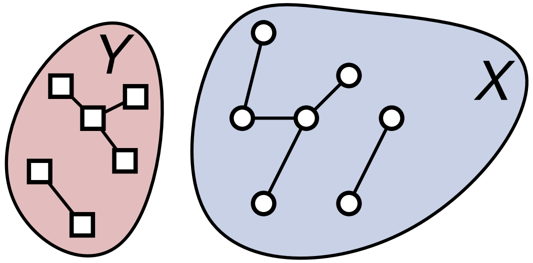


Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch

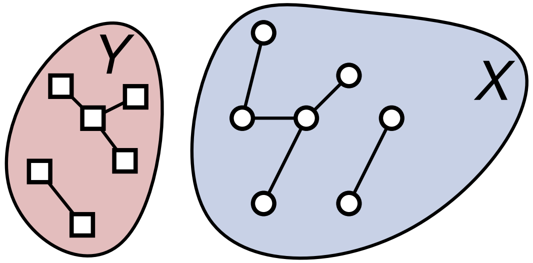


(sonst gibt es keine Lösung)

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation



- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch

(sonst gibt es keine Lösung)

Verzweigungsregel

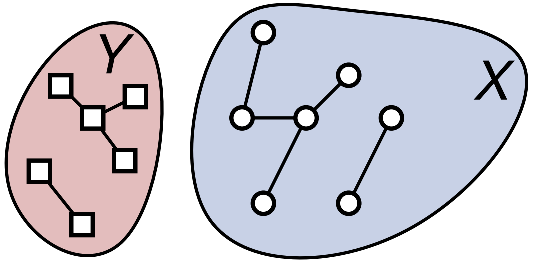
- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

Verzweigungsregel

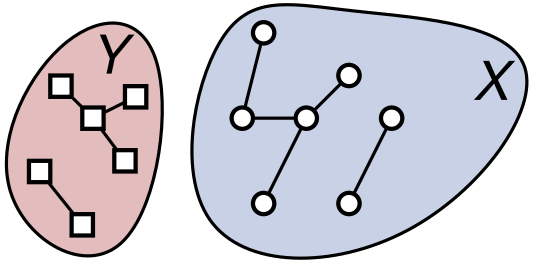
- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$

Möglichkeit 1

- v zu wählen verringert k um 1
- liefert potentiell einen Baum mit beschränkter Höhe

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?



Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch

(sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$

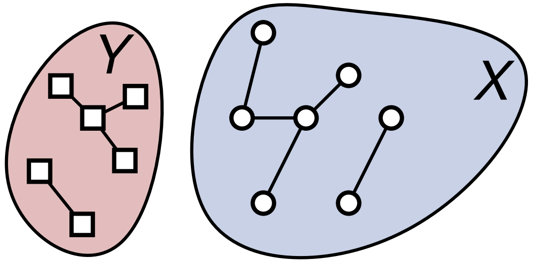
Möglichkeit 2

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

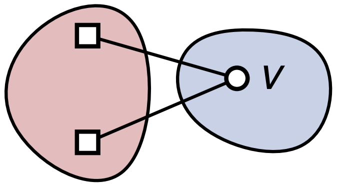
- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$



Möglichkeit 2

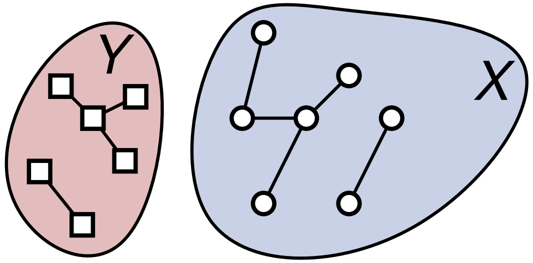
- Idee: nimm an, dass v wenigstens zwei Nachbarn in Y hat

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

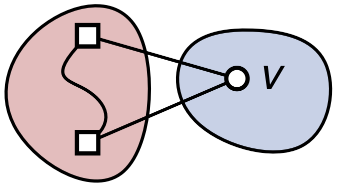
- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$



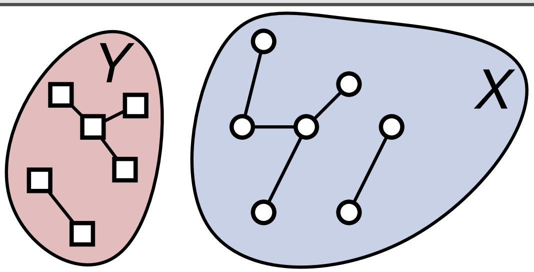
Möglichkeit 2

- Idee: nimm an, dass v wenigstens zwei Nachbarn in Y hat
- Fall 1: Nachbarn sind verbunden $\Rightarrow v$ muss gewählt werden
  $\Rightarrow$ Möglichkeit 2 kann ignoriert werden

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

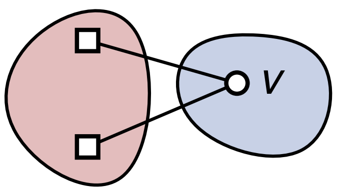
Situation



- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow$ $G[Y]$ ist azyklisch

(sonst gibt es keine Lösung)

Verzweigungsregel



- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$

Möglichkeit 2

- Idee: nimm an, dass v wenigstens zwei Nachbarn in Y hat
- Fall 2: Nachbarn sind nicht verbunden $\Rightarrow$ v verringert die Anzahl der Komponenten in Y um 1

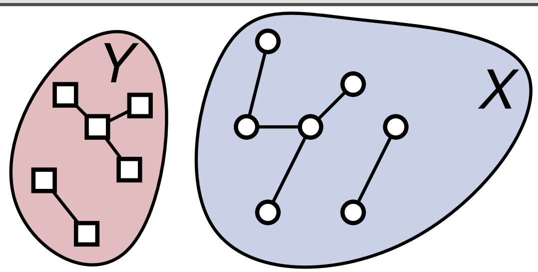
(das kann höchstens k Mal passieren)

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

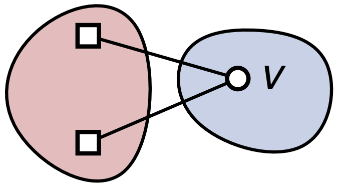
- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$



Möglichkeit 2

- Idee: nimm an, dass v wenigstens zwei Nachbarn in Y hat
- Fall 2: Nachbarn sind nicht verbunden $\Rightarrow$ v verringert die Anzahl der Komponenten in Y um 1 (das kann höchstens k Mal passieren)

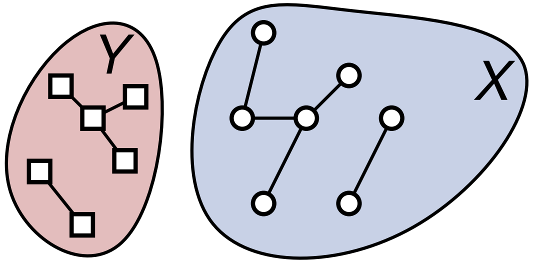
Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

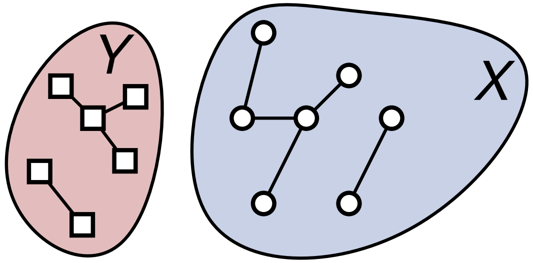
Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

Ein Algorithmus für DISJOINT FVS

Problem: DISJOINT FEEDBACK VERTEX SET (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch

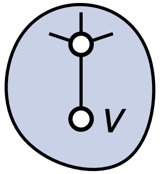


(sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

Warum geht das?

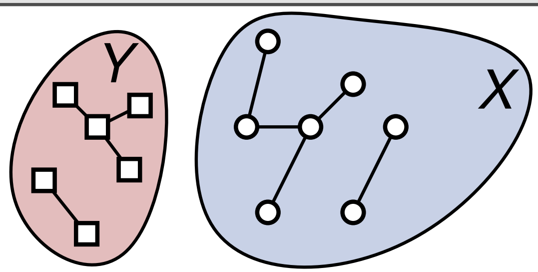


Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



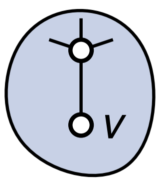
(sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

Warum geht das?

- Fall 1: v hat keinen Nachbar in Y

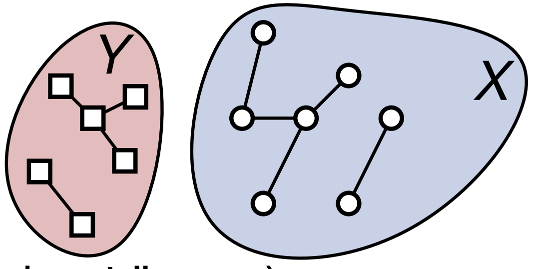


Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow$ $G[Y]$ ist azyklisch



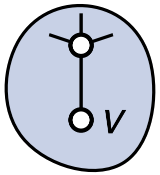
(sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

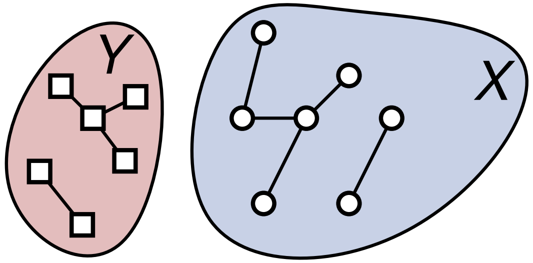
Warum geht das?

- Fall 1: v hat keinen Nachbar in Y
  - v liegt in G auf keinem Kreis und braucht daher nicht gewählt zu werden



Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation



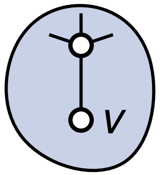
- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch (sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

Warum geht das?

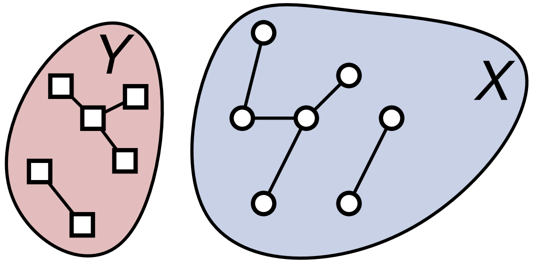
- Fall 1: v hat keinen Nachbar in Y
  - v liegt in G auf keinem Kreis und braucht daher nicht gewählt zu werden
  - **Reduktionsregel:** lösche v



Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation



- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch

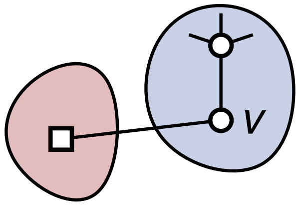
(sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

Warum geht das?

- Fall 1: v hat keinen Nachbar in Y
  - v liegt in G auf keinem Kreis und braucht daher nicht gewählt zu werden
  - **Reduktionsregel:** lösche v

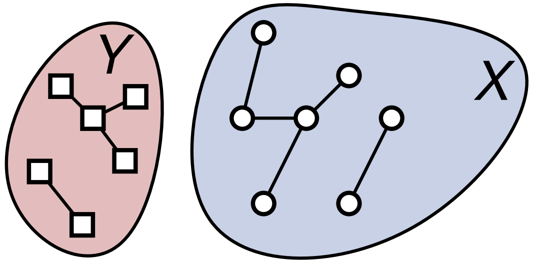


- Fall 2: v hat nur einen Nachbarn in Y

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

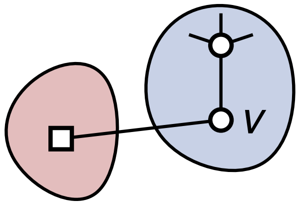


- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch (sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

Warum geht das?

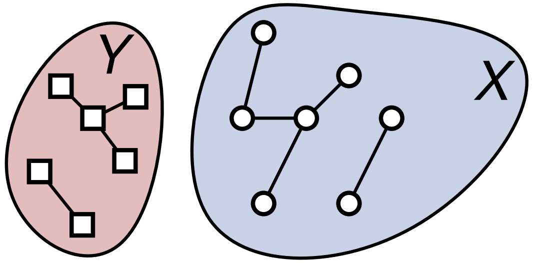


- Fall 1: v hat keinen Nachbar in Y
  - v liegt in G auf keinem Kreis und braucht daher nicht gewählt zu werden
  - **Reduktionsregel:** lösche v
- Fall 2: v hat nur einen Nachbarn in Y
  - v hat Grad 2 in $G \Rightarrow$ jeder Kreis durch v enthält seine Nachbarn

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation



- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch

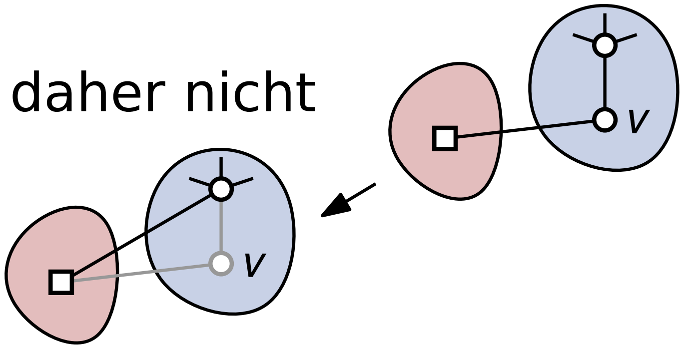
(sonst gibt es keine Lösung)

Zeige: wir finden immer einen Knoten, der $\geq$ zwei Nachbarn in Y hat

- wähle $v \in X$ so, dass v in X nur einen Nachbar hat

Warum geht das?

- Fall 1: v hat keinen Nachbar in Y
  - v liegt in G auf keinem Kreis und braucht daher nicht gewählt zu werden
  - **Reduktionsregel:** lösche v



- Fall 2: v hat nur einen Nachbarn in Y
  - v hat Grad 2 in $G \Rightarrow$ jeder Kreis durch v enthält seine Nachbarn
  - **Reduktionsregel:** ersetzte v durch eine Kante

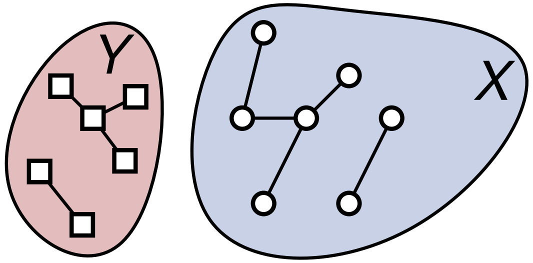
Ein Algorithmus für Disjoint FVS

Karlsruher Institut für Technologie

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$ **(mit mindestens zwei Nachbarn in Y)**
- Möglichkeit 1: wähle $v \in F$
- Möglichkeit 2: wähle $v \notin F$

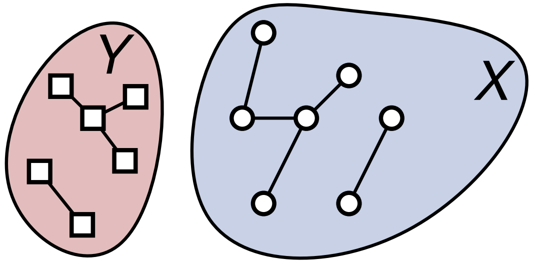
Laufzeit: ??

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch (sonst gibt es keine Lösung)



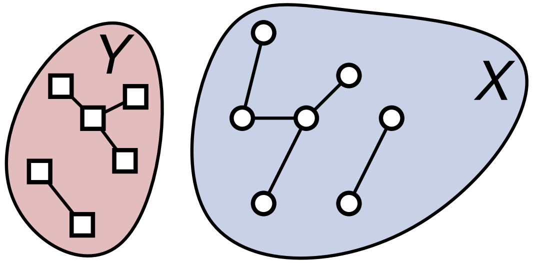
Verzweigungsregel

- betrachte einen Knoten $v \in X$ **(mit mindestens zwei Nachbarn in Y)**
- Möglichkeit 1: wähle $v \in F$ **(k wird kleiner)**
- Möglichkeit 2: wähle $v \notin F$

Laufzeit: ??

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?



Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow G[Y]$ ist azyklisch (sonst gibt es keine Lösung)

Verzweigungsregel

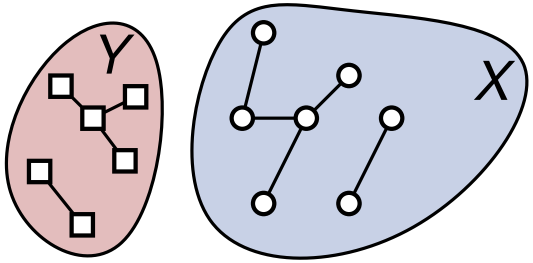
- betrachte einen Knoten $v \in X$ **(mit mindestens zwei Nachbarn in Y)**
- Möglichkeit 1: wähle $v \in F$ **(k wird kleiner)**
- Möglichkeit 2: wähle $v \notin F$ **(Anzahl Komponenten in Y wird kleiner)**

Laufzeit: ??

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation



- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow$ $G[Y]$ ist azyklisch (sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$ **(mit mindestens zwei Nachbarn in Y)**
- Möglichkeit 1: wähle $v \in F$ **(k wird kleiner)**
- Möglichkeit 2: wähle $v \notin F$ **(Anzahl Komponenten in Y wird kleiner)**

Laufzeit

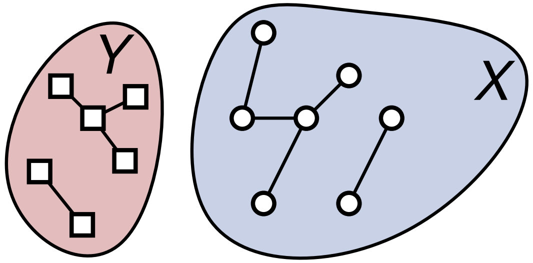
- Höhe des Baumes: maximal $2k$
- $\Rightarrow 4^k \cdot n^{O(1)}$

Ein Algorithmus für Disjoint FVS

Problem: Disjoint Feedback Vertex Set (ungerichtet)
Gegeben sei ein ungerichteter Graph $G = (V, E)$, ein Parameter k und ein FVS Y der Größe $k+1$. Gibt es ein FVS F der Größe k, sodass $F \cap Y = \emptyset$?

Situation

- wähle Knoten aus $X = V \setminus Y$ und keine aus Y
- Y ist ein FVS $\Rightarrow$ $G[X]$ ist azyklisch
- X ist ein FVS $\Rightarrow$ $G[Y]$ ist azyklisch



(sonst gibt es keine Lösung)

Verzweigungsregel

- betrachte einen Knoten $v \in X$ **(mit mindestens zwei Nachbarn in Y)**
- Möglichkeit 1: wähle $v \in F$ **(k wird kleiner)**
- Möglichkeit 2: wähle $v \notin F$ **(Anzahl Komponenten in Y wird kleiner)**

Laufzeit

- Höhe des Baumes: maximal $2k$
- $\Rightarrow 4^k \cdot n^{O(1)}$

Theorem
Feedback Vertex Set (ungerichtet) kann in $5^k \cdot n^{O(1)}$ Zeit gelöst werden.

Zusammenfassung

Theorem
Feedback Vertex Set kann auf Turniergraphen in $O(2^k \cdot n^3)$ Zeit gelöst werden.

Theorem
Feedback Vertex Set (ungerichtet) kann in $5^k \cdot n^{O(1)}$ Zeit gelöst werden.

Zusammenfassung

Theorem
FEEDBACK VERTEX SET kann auf Turniergraphen in $O(2^k \cdot n^3)$ Zeit gelöst werden.

Theorem
FEEDBACK VERTEX SET (ungerichtet) kann in $5^k \cdot n^{O(1)}$ Zeit gelöst werden.

Technik: Iterative Kompression

- zeige: (PROBLEM) lösbar $\Leftarrow$ (PROBLEM)-COMPRESSION lösbar
- zeige: (PROBLEM)-COMPRESSION lösbar $\Leftarrow$ DISJOINT-(PROBLEM) lösbar
- löse: DISJOINT-(PROBLEM) (das ist der schwierige Teil!)

Zusammenfassung

Theorem
FEEDBACK VERTEX SET kann auf Turniergraphen in $O(2^k \cdot n^3)$ Zeit gelöst werden.

Theorem
FEEDBACK VERTEX SET (ungerichtet) kann in $5^k \cdot n^{O(1)}$ Zeit gelöst werden.

Technik: Iterative Kompression

- zeige: (PROBLEM) lösbar $\Leftarrow$ (PROBLEM)-COMPRESSION lösbar
- zeige: (PROBLEM)-COMPRESSION lösbar $\Leftarrow$ DISJOINT-(PROBLEM) lösbar
- löse: DISJOINT-(PROBLEM) (das ist der schwierige Teil!)

Was macht DISJOINT-(PROBLEM) leichter als PROBLEM?

Zusammenfassung

Theorem
Feedback Vertex Set kann auf Turniergraphen in $O(2^k \cdot n^3)$ Zeit gelöst werden.

Theorem
Feedback Vertex Set (ungerichtet) kann in $5^k \cdot n^{O(1)}$ Zeit gelöst werden.

Technik: Iterative Kompression

- zeige: (Problem) lösbar $\Leftarrow$ (Problem)-Compression lösbar
- zeige: (Problem)-Compression lösbar $\Leftarrow$ Disjoint-(Problem) lösbar
- löse: Disjoint-(Problem) (das ist der schwierige Teil!)

Was macht Disjoint-(Problem) leichter als Problem?

- Lösung Y der Größe $k + 1$ ist bekannt
- neue Lösung muss disjunkt sein zu Y
- Komplement von Y ist eine Lösung