Travelling Salesman Problem

Amelie Koch und Josua Kugler

CdE Cyberaka 2020

Motivation

- Kürzeste Rundreise finden.

Motivation

- Kürzeste Rundreise finden.
  - Handlungsreisende

Motivation

Abbildung: Optimaler Reiseweg eines Handlungsreisenden durch die 15 größten Städte Deutschlands. Die angegebene Route ist die kürzeste von 43.589.145.600 möglichen.

Motivation

- Kürzeste Rundreise finden.
  - Handlungsreisende
  - Computerchips

Motivation

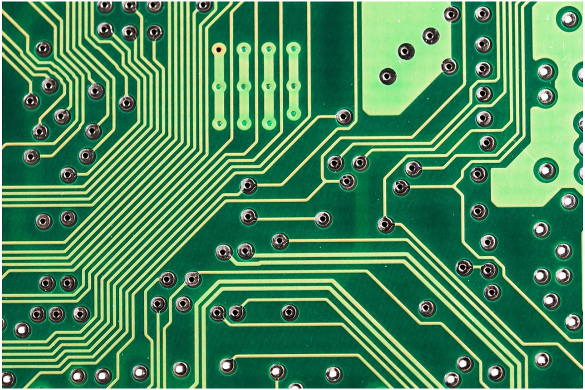


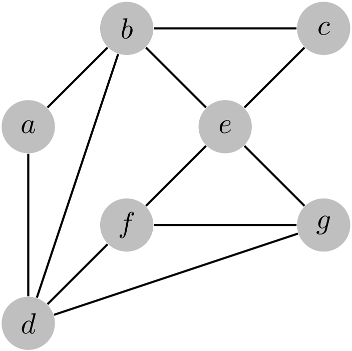
Abbildung: Ein großer Teil des Aufwands beim Erstellen vom Computerchips entfällt auf Optimierung, z.B. auch die kürzeste Route zum Bohren zu finden.

Motivation

- Kürzeste Rundreise finden.
    - Handlungsreisende
    - Computerchips

Grundlegende Definitionen

Sei $G = (V, E)$ ein Graph mit Knotenmenge V und Kantenmenge E.

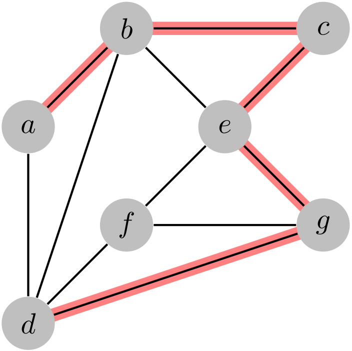


Grundlegende Definitionen

Sei $G = (V, E)$ ein Graph mit Knotenmenge V und Kantenmenge E.

Definition (Pfad)

$v_0, v_1, \ldots, v_n \in V$ mit Kanten
$e = (v_{i-1}, v_i) \in E \quad \forall 1 \leq i \leq n$



Grundlegende Definitionen

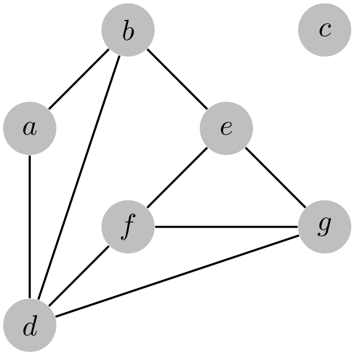
Sei $G = (V, E)$ ein Graph mit
Knotenmenge V und Kantenmenge E.

Definition (zusammenhängend)

$\forall v_1, v_2 \in V \exists$ Pfad von v_1 nach v_2.

Definition (Zusammenhangskomponenten)

maximal zusammenhängende Teilgraphen



Grundlegende Definitionen

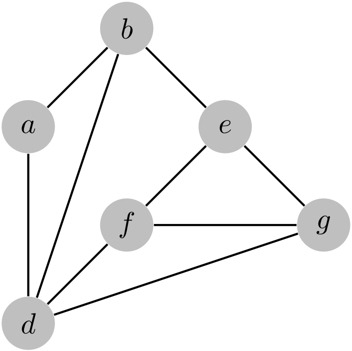
Sei $G = (V, E)$ ein Graph mit Knotenmenge V und Kantenmenge E.

Definition (zusammenhängend)

$\forall v_1, v_2 \in V \exists$ Pfad von v_1 nach v_2.

Definition (Zusammenhangskomponenten)

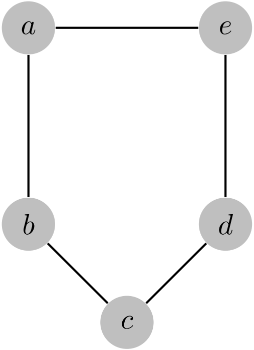
maximal zusammenhängende Teilgraphen



Grundlegende Definitionen

Definition (Kreis)

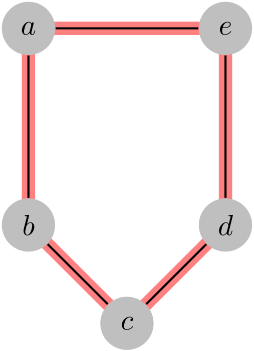
Ist $(v_1, \ldots, v_{n-1})$ mit $n > 1$ ein Pfad, so heißt $(v_0, \ldots, v_{n-1}, v_0)$ Kreis.



Grundlegende Definitionen

Definition (Kreis)

Ist $(v_1, \ldots, v_{n-1})$ mit $n > 1$ ein Pfad, so heißt $(v_0, \ldots, v_{n-1}, v_0)$ Kreis.



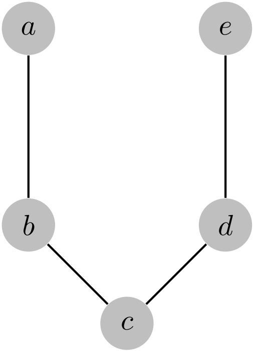
Grundlegende Definitionen

Definition (Kreis)

Ist $(v_1, \ldots, v_{n-1})$ mit $n > 1$ ein Pfad, so heißt $(v_0, \ldots, v_{n-1}, v_0)$ Kreis.

Definition

Einen Graphen ohne Kreise nennt man **azyklisch**.



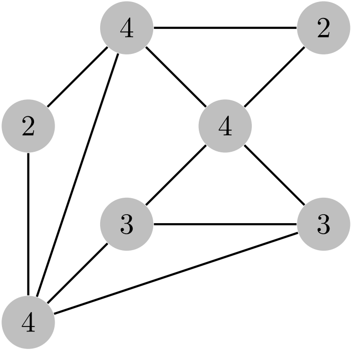
Grundlegende Definitionen

Definition (Grad)

$d(v) = |\{e \in E | v \in e\}|$

Lemma

Die Anzahl der Knoten mit ungeradem Grad ist gerade.



Definition Baum

Definition

Ein Baum ist eine ausdauernde und verholzende Samenpflanze mit einer dominierende Sprossachse, die durch sekundäres Dickenwachstum an Umfang zunimmt.

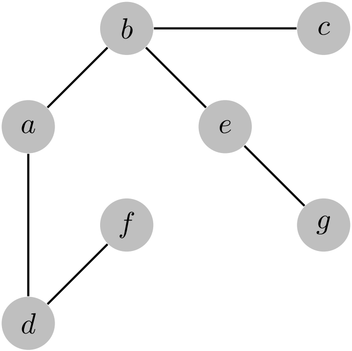
Definition Baum

Nein Spass, jedes Kind weiß, dass das so nicht stimmt. In Wirklichkeit ist das ganz anders.

Definition Baum

Definition (Baum)

zusammenhängender, azyklischer Graph



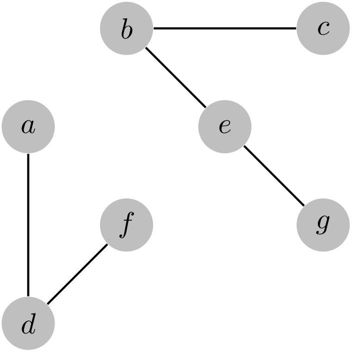
Definition Baum

Definition (Baum)

zusammenhängender, azyklischer Graph

Definition (Wald)

Graph, dessen Zusammenhangskomponenten Bäume sind



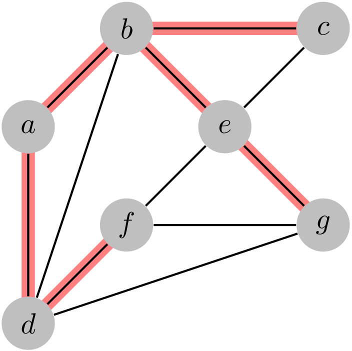
Definition Baum

Definition (Baum)

zusammenhängender, azyklischer Graph

Definition (aufspannender Baum)

Ein **aufspannender Baum** eines Graphen G ist ein Baum, der alle Knoten von G enthält.



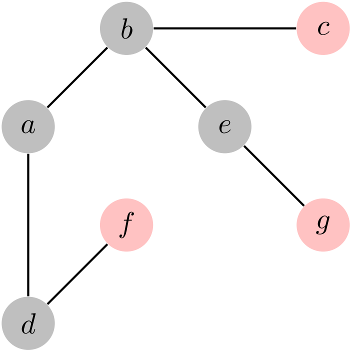
Definition Baum

Definition (Baum)

zusammenhängender, azyklischer Graph

Definition (Blatt)

Ein Knoten eines Baums mit Grad 1 heißt **Blatt**.



Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

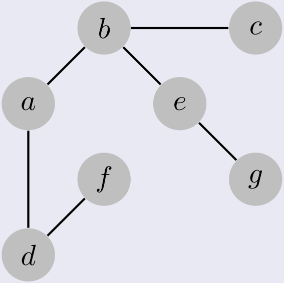
Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Beweis.

- Zusammenhang $\implies \exists$ Pfad.



□

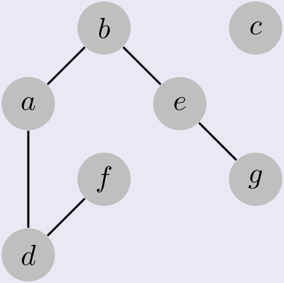
Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Beweis.

- Zusammenhang $\implies \exists$ Pfad.



$\square$

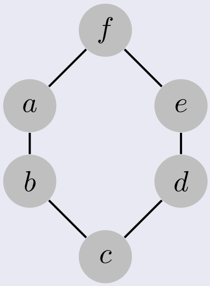
Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Beweis.

- Zusammenhang $\implies \exists$ Pfad.
- Annahme: Es gibt verschiedene Pfade zwischen v_1 und v_2 $\implies$ Kreis $\implies$ ↯.



□

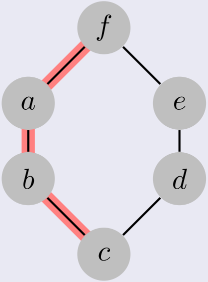
Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Beweis.

- Zusammenhang $\implies \exists$ Pfad.
- Annahme: Es gibt verschiedene Pfade zwischen v_1 und v_2 $\implies$ Kreis $\implies$ ↯.



□

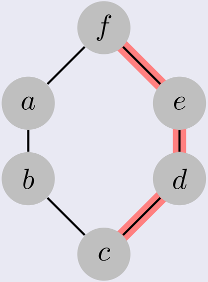
Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Beweis.

- Zusammenhang $\implies \exists$ Pfad.
- Annahme: Es gibt verschiedene Pfade zwischen v_1 und v_2 $\implies$ Kreis $\implies$ ↯.



□

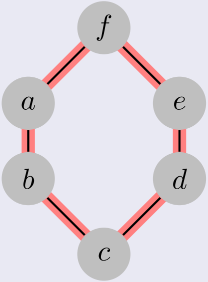
Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Beweis.

- Zusammenhang $\implies \exists$ Pfad.
- Annahme: Es gibt verschiedene Pfade zwischen v_1 und v_2 $\implies$ Kreis $\implies \lightning$.



$\square$

Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

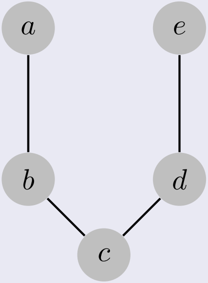
Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Beweis.

- Kante entfernen



$\square$

Grundlegende Eigenschaften eines Baums

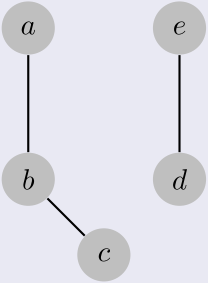
Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Beweis.

- Kante entfernen

$\Longrightarrow$ Pfad unterbrochen



$\square$

Grundlegende Eigenschaften eines Baums

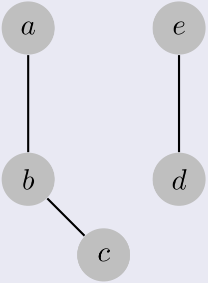
Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Beweis.

- Kante entfernen

$\Longrightarrow$ Pfad unterbrochen
$\overset{\text{Lemma 1}}{\Longrightarrow}$ nicht mehr zusammenhängend



□

Grundlegende Eigenschaften eines Baums

Lemma

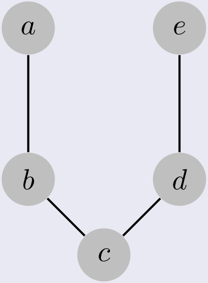
Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Beweis.

- Kante entfernen

$\Longrightarrow$ Pfad unterbrochen
$\overset{\text{Lemma 1}}{\Longrightarrow}$ nicht mehr zusammenhängend

- Kante hinzufügen



□

Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Beweis.

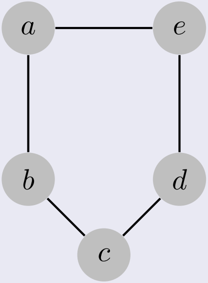
- Kante entfernen

  $\Longrightarrow$ Pfad unterbrochen

  $\overset{\text{Lemma 1}}{\Longrightarrow}$ nicht mehr zusammenhängend

- Kante hinzufügen

  $\Longrightarrow$ neuer Weg



$\square$

Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Beweis.

- Kante entfernen

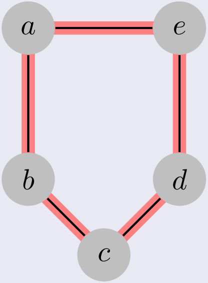
  $\Longrightarrow$ Pfad unterbrochen

  $\overset{\text{Lemma 1}}{\Longrightarrow}$ nicht mehr zusammenhängend

- Kante hinzufügen

  $\Longrightarrow$ neuer Weg

  $\overset{\text{Lemma 1}}{\Longrightarrow}$ Kreis



$\square$

Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum mit n Knoten besitzt $n-1$ Kanten.

Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum mit n Knoten besitzt $n-1$ Kanten.

Beweis.

- Induktionsanfang:
  1 Knoten $\implies$ 0 Kanten

$\square$

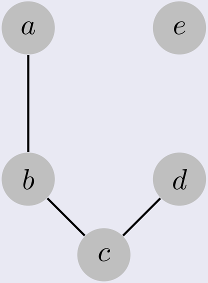
Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum mit n Knoten besitzt $n-1$ Kanten.

Beweis.

- Induktionsanfang:
  1 Knoten $\implies$ 0 Kanten
- Induktionsschritt:
  $k \to k+1$ Knoten $\implies$ $+1$ Kante, sonst Kreis.



$\square$

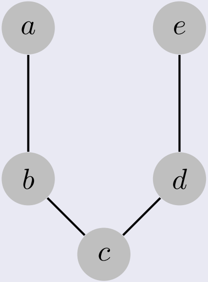
Grundlegende Eigenschaften eines Baums

Lemma

Ein Baum mit n Knoten besitzt $n-1$ Kanten.

Beweis.

- Induktionsanfang:
  1 Knoten $\implies$ 0 Kanten
- Induktionsschritt:
  $k \to k+1$ Knoten $\implies$ $+1$ Kante, sonst Kreis.



□

Grundlegende Eigenschaften eines Baums

Lemma

In einem Baum gibt es genau einen Pfad zwischen zwei Knoten.

Lemma

Ein Baum ist minimal zusammenhängend und maximal kreisfrei.

Lemma

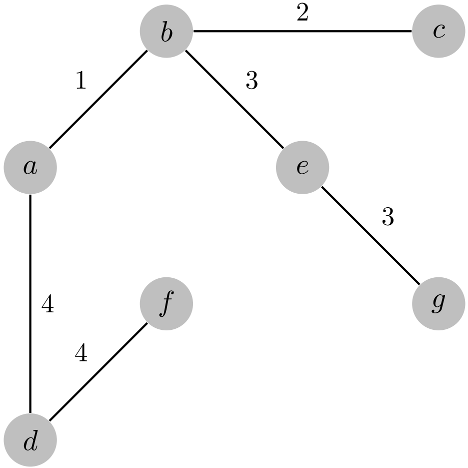
Ein Baum mit n Knoten besitzt $n - 1$ Kanten.

Allgemeine Definitionen

Definition

Ein Graph $G = (V, E)$ mit einer Funktion $w : E \to \mathbb{R}$ heißt gewichteter Graph.

Allgemeine Definitionen



Allgemeine Definitionen

Definition

Ein Graph $G = (V, E)$ mit einer Funktion $w : E \to \mathbb{R}$ heißt gewichteter Graph.

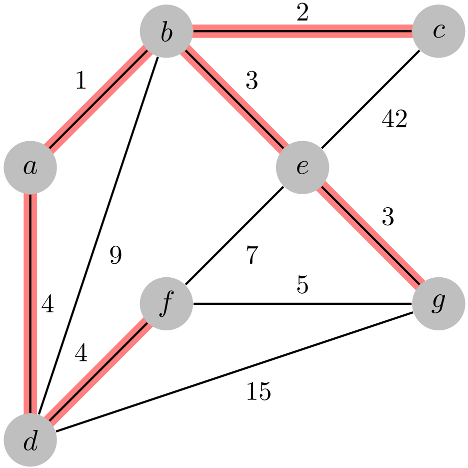
Definition

Der minimale Spannbaum $G' = (V, E')$ eines gewichteten, zusammenhängenden Graphen $G = (V, E)$ ist ein aufspannender Baum, für den

$$\sum_{e \in E'} w(e)$$

minimal ist.

Allgemeine Definitionen



Algorithmus von Kruskal

Eingabe: gewichteter Graph $G = (V, E)$, Funktion $w : E \to \mathbb{R}$

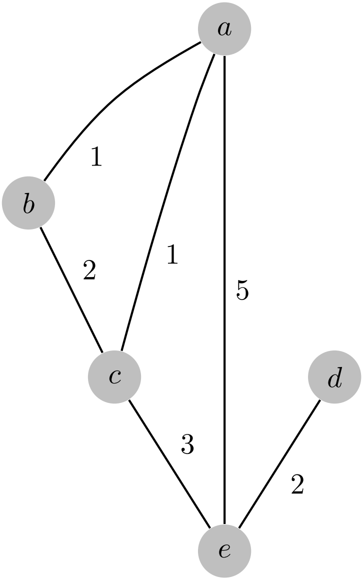
Ausgabe: minimaler Spannbaum $G'(V, E')$ von G

(1) **while** $|E'| < |V| - t$

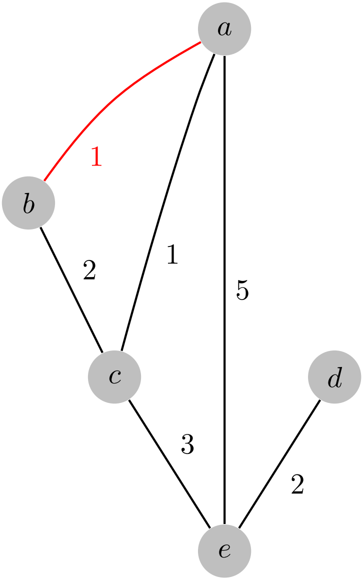
(2) betrachte Kante e aus G mit $w(e) = \min_{e \in E} w(e)$

(3) **if** G' mit e azyklisch **then** e von G zu G'

(4) **else** entferne e in G



$$G' = (V, \emptyset)$$



while $|E'| < |V| - 1$

betrachte Kante e aus G mit

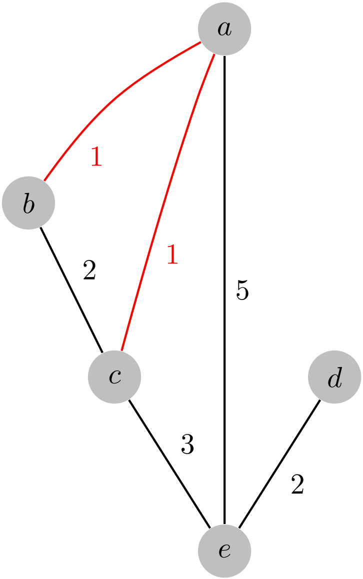
$$w(e) = \min_{e \in E} w(e),$$

Setze

$$E = E \setminus \{e\}$$

if G' mit e azyklisch

$$E' = E' \cup \{e\}$$



while $|E'| < |V| - 1$

betrachte Kante e aus G mit

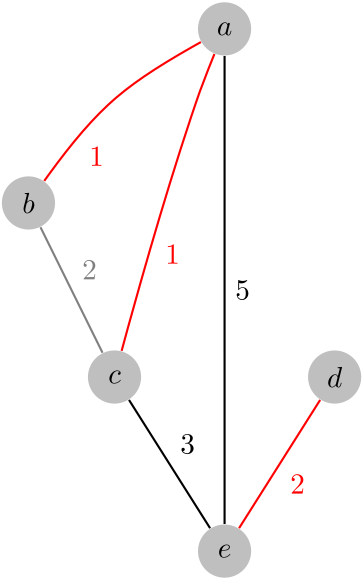
$$w(e) = \min_{e \in E} w(e),$$

Setze

$$E = E \setminus \{e\}$$

if G' mit e azyklisch

$$E' = E' \cup \{e\}$$



while $|E'| < |V| - 1$

betrachte Kante e aus G mit

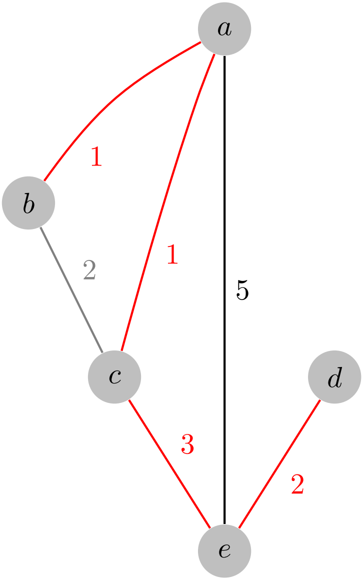
$$w(e) = \min_{e \in E} w(e),$$

Setze

$$E = E \setminus \{e\}$$

if G' mit e azyklisch

$$E' = E' \cup \{e\}$$



while $|E'| < |V| - 1$

betrachte Kante e aus G mit

$$w(e) = \min_{e \in E} w(e),$$

Setze

$$E = E \setminus \{e\}$$

if G' mit e azyklisch

$$E' = E' \cup \{e\}$$

Grundlegende Definitionen

Sei S eine endliche Menge und $U \subseteq P(S)$ Familie von Teilmengen.

Grundlegende Definitionen

Sei S eine endliche Menge und $U \subseteq P(S)$ Familie von Teilmengen.

Definition

Das Paar $M = (S, U)$ heißt **Matroid** und U die Familie der **unabhängigen Mengen** von M, wenn gilt:

1. $\emptyset \in U$
2. $A \in U, B \subseteq A \implies B \in U$
3. $A, B \in U,\ \#B = \#A + 1 \implies \exists v \in B \setminus A$ mit $A \cup \{v\} \in U$

Grundlegende Definitionen

Sei S eine endliche Menge und $U \subseteq P(S)$ Familie von Teilmengen.

Definition

Das Paar $M = (S, U)$ heißt **Matroid** und U die Familie der **unabhängigen Mengen** von M, wenn gilt:

1. $\emptyset \in U$
2. $A \in U, B \subseteq A \implies B \in U$
3. $A, B \in U,\ \#B = \#A + 1 \implies \exists v \in B \setminus A$ mit $A \cup \{v\} \in U$

Definition

Eine maximale unabhängige Menge heißt eine **Basis** des Matroids. Alle Basen enthalten die gleiche Anzahl von Elementen, der **Rang** $r(M)$ des Matroids.

Matroide und Vektorräume

Begriffsverwandtschaft mit Matrix:

Matroide und Vektorräume

Begriffsverwandtschaft mit Matrix:

- $S \cong$ Menge der Spaltenvektoren einer Matrix.

Matroide und Vektorräume

Begriffsverwandtschaft mit Matrix:

- $S \cong$ Menge der Spaltenvektoren einer Matrix.
- $U \cong$ Menge der linear unabhängigen Systeme bestehend aus Vektoren $\in S$.

Matroide und Vektorräume

Begriffsverwandtschaft mit Matrix:

- $S \cong$ Menge der Spaltenvektoren einer Matrix.
- $U \cong$ Menge der linear unabhängigen Systeme bestehend aus Vektoren $\in S$.
- Basis $\cong$ maximal linear unabhängiges System (Basis des Bildes der Matrix).

Matroide und Vektorräume

Begriffsverwandtschaft mit Matrix:

- $S \cong$ Menge der Spaltenvektoren einer Matrix.
- $U \cong$ Menge der linear unabhängigen Systeme bestehend aus Vektoren $\in S$.
- Basis $\cong$ maximal linear unabhängiges System (Basis des Bildes der Matrix).
- Rang $\cong$ Anzahl der Vektoren einer Basis, Rang der Matrix.

Matroide und Vektorräume

$$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$S = \{ \overbrace{\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}}^{a}, \overbrace{\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}}^{b}, \overbrace{\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}}^{c}, \overbrace{\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}}^{d} \}$$

$$\begin{aligned} U = \{ & \emptyset, \{a\}, \{b\}, \{c\}, \{d\}, \{a,b\}, \\ & \{a,c\}, \{a,d\}, \{b,c\}, \{b,d\}, \{c,d\}, \\ & \{a,b,c\}, \{a,c,d\}, \{b,c,d\} \} \end{aligned}$$

$$\underline{e} = \{a, b, c\}$$

$$\mathrm{rang}(S, U) = 3$$

Matroide und Graphen

Sei $W \subseteq P(E)$ die Familie der Kantenmenge aller Wälder eines Graphen $G = (V, E)$.

Lemma

Ist $G = (V, E)$ Graph, so ist $M = (E, W)$ ein Matroid.

Matroide und Graphen

Sei $W \subseteq P(E)$ die Familie der Kantenmenge aller Wälder eines Graphen $G = (V, E)$.

Lemma

Ist $G = (V, E)$ Graph, so ist $M = (E, W)$ ein Matroid.

Beweis.

- Axiom 1

□

Matroide und Graphen

Sei $W \subseteq P(E)$ die Familie der Kantenmenge aller Wälder eines Graphen $G = (V, E)$.

Lemma

Ist $G = (V, E)$ Graph, so ist $M = (E, W)$ ein Matroid.

Beweis.

- Axiom 1
- Axiom 2

$\square$

Matroide und Graphen

Sei $W \subseteq P(E)$ die Familie der Kantenmenge aller Wälder eines Graphen $G = (V, E)$.

Lemma

Ist $G = (V, E)$ Graph, so ist $M = (E, W)$ ein Matroid.

Beweis.

- Wälder $W = (V, A)$, $W' = (V, B)$ mit $\#B = \#A + 1$.
  Zusammenhangskomponenten $T_1, ..., T_m$, Eckenmengen $V_1, ..., V_m$,
  Kantenmengen $A_1, ..., A_m$.

Matroide und Graphen

Sei $W \subseteq P(E)$ die Familie der Kantenmenge aller Wälder eines Graphen $G = (V, E)$.

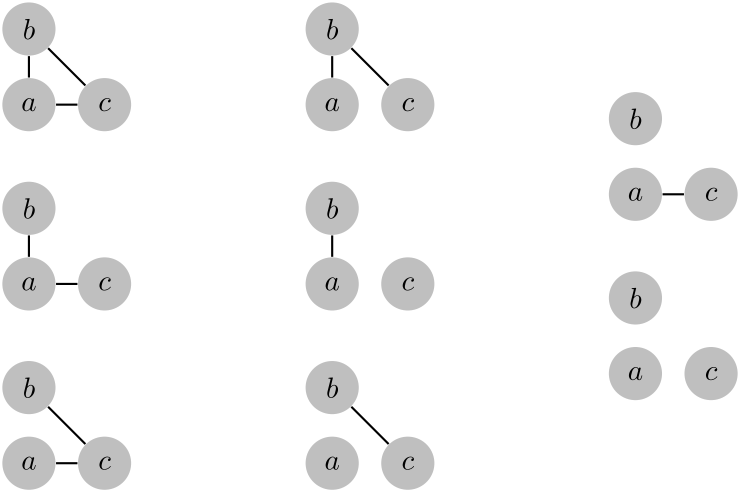
Lemma

Ist $G = (V, E)$ Graph, so ist $M = (E, W)$ ein Matroid.

Beweis.

- Wälder $W = (V, A)$, $W' = (V, B)$ mit $\#B = \#A + 1$.
  Zusammenhangskomponenten $T_1, ..., T_m$, Eckenmengen $V_1, ..., V_m$,
  Kantenmengen $A_1, ..., A_m$.
- Nun: $\#A_i = \#V_i - 1$, $V = V_1 \cup ... \cup V_m$, $A = A_1 \cup ... \cup A_m$.
  $\#B > \#A \implies \exists$ Kante $k \in B$, die V_s, V_t verbindet.
  Dann ist $W'' = (V, A \cup k)$ Wald.

Das Matroid $M = (V, W)$



Matroide und Graphen - Folgerungen

Korollar

Basen von $M = (E, W)$ sind die aufspannenden Wälder.

Matroide und Graphen - Folgerungen

Korollar

Basen von $M = (E, W)$ sind die aufspannenden Wälder.

Korollar

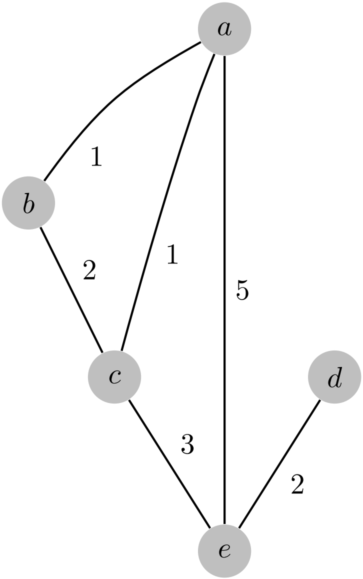
Rang des Matroids ist $r(M) = \#V - t$, wobei t die Anzahl der Komponenten von G ist.

Algorithmus von Kruskal in Matroidsprache

Theorem

$M = (E, W)$ Matroid mit Gewichtsfunktion $w : E \to \mathbb{R}$.
Algorithmus liefert minimalen Spannbaum:

1. *Sei $A_0 = \emptyset \in W$.*
2. *Ist $A_i = \{a_1, ..., a_i\} \subseteq E$, so sei $X_i = \{e \in E \setminus A_i \mid A_i \cup \{e\} \in W\}$. Falls $X_i = \emptyset$, so ist A_i gesuchte Basis. Andernfalls wähle ein $a_{i+1} \in X_i$ mit minimalem Gewicht, und setze $A_{i+1} = A_i \cup \{a_{i+1}\}$. Iteriere (2).*



$M(E, W)$ Matroid, w Gewichtung.

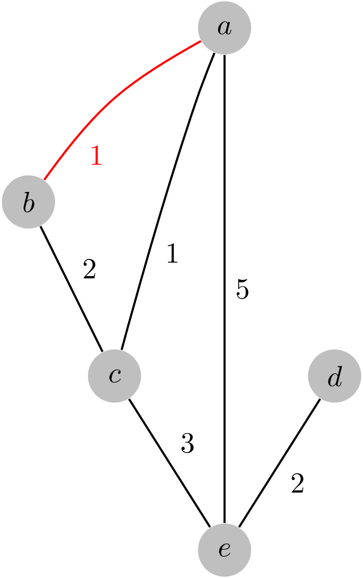
$A_0 = \emptyset$, $X_0 = E$.

In unserem Beispiel:

$i = 0$

$A_0 = \emptyset$

$X_0 =$
$\{(a, b), (b, c), (a, c), (a, e), (c, e), (e, d)\}$



while $X_i \neq \emptyset$

$a_i \in X_{i-1}$ mit minimalem Gewicht.

$A_i = A_{i-1} \cup \{a_i\}$

$X_i = \{e \in E \setminus A_i | A_i \cup \{e\} \in W\}$

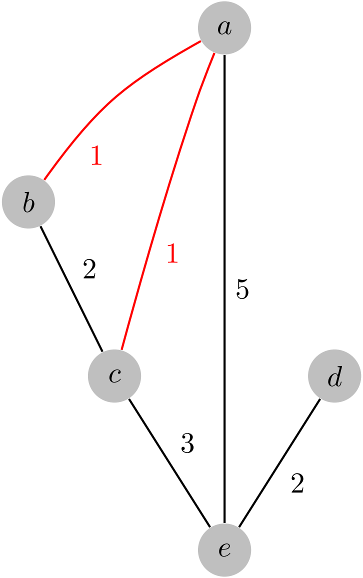
$i{+}{+}$

In unserem Beispiel:

$i = 1$

$A_1 = \{(a, b)\}$

$X_1 = \{(b, c), (a, c), (a, e), (c, e), (e, d)\}$



while $X_i \neq \emptyset$

$a_i \in X_{i-1}$ mit minimalem Gewicht.

$A_i = A_{i-1} \cup \{a_i\}$

$X_i = \{e \in E \setminus A_i | A_i \cup \{e\} \in W\}$

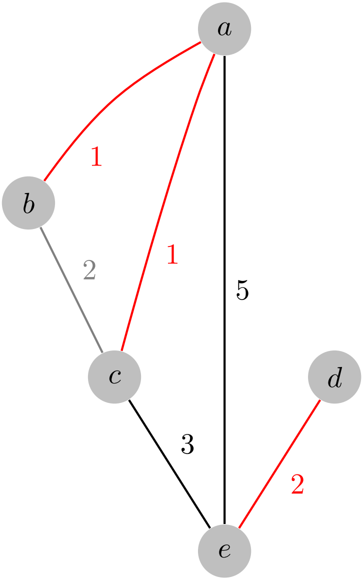
$i++$

In unserem Beispiel:

$i = 2$

$A_2 = \{(a,b),(a,c)\}$

$X_2 = \{(a,e),(c,e),(e,d)\}$



while $X_i \neq \emptyset$

$a_i \in X_{i-1}$ mit minimalem Gewicht.

$A_i = A_{i-1} \cup \{a_i\}$

$X_i = \{e \in E \setminus A_i | A_i \cup \{e\} \in W\}$

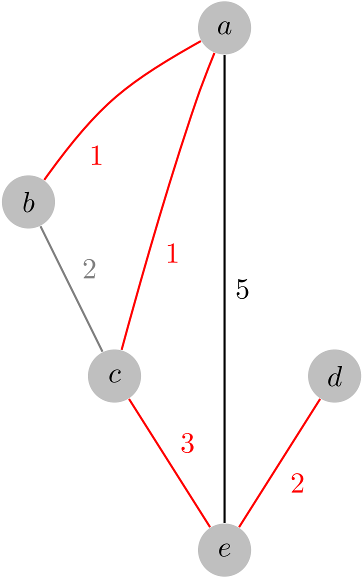
$i++$

In unserem Beispiel:

$i = 3$

$A_3 = \{(a,b),(a,c),(e,d)\}$

$X_3 = \{(a,e),(c,e)\}$



while $X_i \neq \emptyset$

$a_i \in X_{i-1}$ mit minimalem Gewicht.

$A_i = A_{i-1} \cup \{a_i\}$

$X_i = \{e \in E \setminus A_i | A_i \cup \{e\} \in W\}$

$i++$

In unserem Beispiel:

$i = 4$

$A_4 = \{(a,b),(a,c),(e,d),(c,e)\}$

$X_4 = \emptyset$

Greedy-Algorithmus auf gewichtetem Matroid - Korrektheitsbeweis

Beweis.

- Sei $A = \{a_1, ..., a_r\}$ die erhaltene Menge.

$\square$

Greedy-Algorithmus auf gewichtetem Matroid - Korrektheitsbeweis

Beweis.

- Sei $A = \{a_1, ..., a_r\}$ die erhaltene Menge.
- Axiom 3 $\implies$ A ist Basis.

$\square$

Greedy-Algorithmus auf gewichtetem Matroid - Korrektheitsbeweis

Beweis.

- Sei $A = \{a_1, ..., a_r\}$ die erhaltene Menge.
- Axiom 3 $\implies$ A ist Basis.
- Konstruktion und Axiom 2 $\implies$ $w(a_1) \leq ... \leq w(a_r)$.

$\square$

Greedy-Algorithmus auf gewichtetem Matroid - Korrektheitsbeweis

Beweis.

- Sei $A = \{a_1, ..., a_r\}$ die erhaltene Menge.
- Axiom 3 $\implies$ A ist Basis.
- Konstruktion und Axiom 2 $\implies$ $w(a_1) \leq ... \leq w(a_r)$.
- Angenommen $B = \{b_1, ..., b_r\}$ wäre eine Basis mit $w(B) < w(A)$ mit $w(b_1) \leq ... \leq w(b_r)$. Dann existiert minimaler Index i mit $w(b_i) < w(a_i)$ wobei $i \geq 2$ gilt.

□

Greedy-Algorithmus auf gewichtetem Matroid - Korrektheitsbeweis

Beweis.

- Sei $A = \{a_1, ..., a_r\}$ die erhaltene Menge.
- Axiom 3 $\implies$ A ist Basis.
- Konstruktion und Axiom 2 $\implies$ $w(a_1) \leq ... \leq w(a_r)$.
- Angenommen $B = \{b_1, ..., b_r\}$ wäre eine Basis mit $w(B) < w(A)$ mit $w(b_1) \leq ... \leq w(b_r)$. Dann existiert minimaler Index i mit $w(b_i) < w(a_i)$ wobei $i \geq 2$ gilt.
- $A_{i-1} = \{a_1, ..., a_{i-1}\}$, $B_i = \{b_1, ..., b_i\}$ unabhängige Mengen. Axiom 3 $\implies$ $\exists\, b_j \in B_i \setminus A_{i-1}$ sodass $A_{i-1} \cup \{b_j\} \in U$. Nun $w(b_j) \leq w(b_i) < w(a_i)$ $\implies$ Widerspruch.

$\square$

Die Heuristik von Christofides

Als Ausgangsgraph betrachten wir $G = K_n$.

1. minimalen Spannbaum T von G
2. kostenminimales, perfektes Matching M auf den Knoten von T mit ungeradem Grad
3. Eulerkreis E auf $M \dot{\cup} T$
4. Hamiltonkreis H aus Eulerkreis E

Die Heuristik von Christofides - Schritt 1

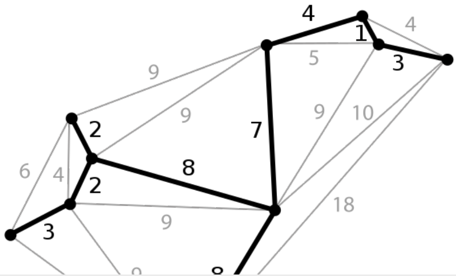
Finde einen minimalen Spannbaum T von G.

- bei uns: Kruskal-Algorithmus,
  $\mathcal{O}(|E|(|E| + |V|)) = \mathcal{O}(\frac{n(n-1)}{2}(\frac{n(n-1)}{2} + n)) = \mathcal{O}(n^4)$.
- in optimaler Implementierung mit Adjazenzmatrix: Prim
  $\mathcal{O}(|V|^2) = \mathcal{O}(n^2)$.
- $w(T) < w(H_{\text{opt}})$

Die Heuristik von Christofides - Schritt 1

Finde einen minimalen Spannbaum T von G.

- bei uns: Kruskal-Algorithmus, $\mathcal{O}(|E|(|E|+|V|)) = \mathcal{O}(\frac{n(n-1)}{2}(\frac{n(n-1)}{2}+n)) = \mathcal{O}(n^4)$.
- in optimaler Implementierung mit Adjazenzmatrix: Prim $\mathcal{O}(|V|^2) = \mathcal{O}(n^2)$.
- $w(T) < w(H_{\text{opt}})$



Die Heuristik von Christofides - Schritt 2

Finde ein minimales perfektes Matching M auf den Knoten von T mit ungeradem Grad.

- Optimal: Blossom-Algorithmus, Linear Programming.
- Greedy-Algorithmus, $\mathcal{O}(|E'|\log|E'|) = \mathcal{O}(n^2\log n)$.
- $w(M) < \frac{1}{2}w(H_{\text{opt}})$.

Die Heuristik von Christofides - Schritt 3

Finde einen Eulerkreis E auf $T \dot{\cup} M$.

- Bei uns: Algorithmus von Fleury $\mathcal{O}(|E_{T \dot{\cup} M}|^2) = \mathcal{O}(n^2)$.
- Alternativ: Algorithmus von Hierholzer $\mathcal{O}(\backslash)$.
- $w(E) \leq w(T + M) = w(T) + w(M) < \frac{3}{2} w(H_{\text{opt}})$.

Die Heuristik von Christofides - Schritt 4

Finde einen Hamiltonkreis H aus E.

- Algorithmus: Überspringe alle bereits besuchten Knoten: $\mathcal{O}(n)$.
- $w(H) \leq w(E) < \frac{3}{2} w(H_{\text{opt}})$.

Die Heuristik von Christofides

Als Ausgangsgraph betrachten wir $G = K_n$.

1. minimalen Spannbaum T von G
2. kostenminimales, perfektes Matching M auf den Knoten von T mit ungeradem Grad
3. Eulerkreis E auf $M \dot{\cup} T$
4. Hamiltonkreis H aus Eulerkreis E

Weitere Optimierungen

- k-opt-Verfahren.

Weitere Optimierungen

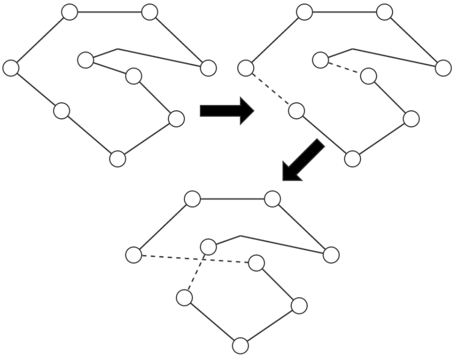


Abbildung: 2-opt-Verfahren

Weitere Optimierungen

- k-opt-Verfahren.
- Ant-Colony-Optimization

Weitere Optimierungen

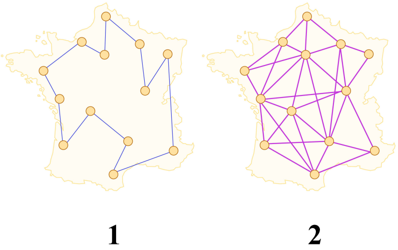


Abbildung: Die Dicke der Kanten ist proportional zu den Pheromonen, die darauf verteilt sind

Weitere Optimierungen

- k-opt-Verfahren.
- Ant-Colony-Optimization
- Andere Ansätze: Linear Programming