Karlsruher Institut für Technologie

Algorithmen II
Vorlesung am 19.11.2013

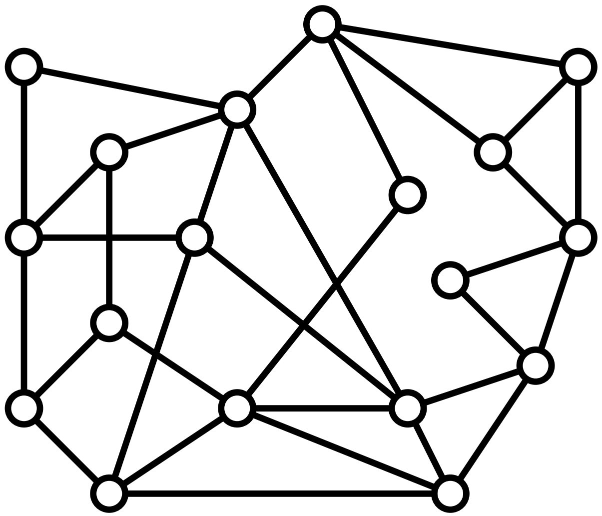
Kreisbasen, Matroide & Algorithmen

INSTITUT FÜR THEORETISCHE INFORMATIK · PROF. DR. DOROTHEA WAGNER

Kreisbasen

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

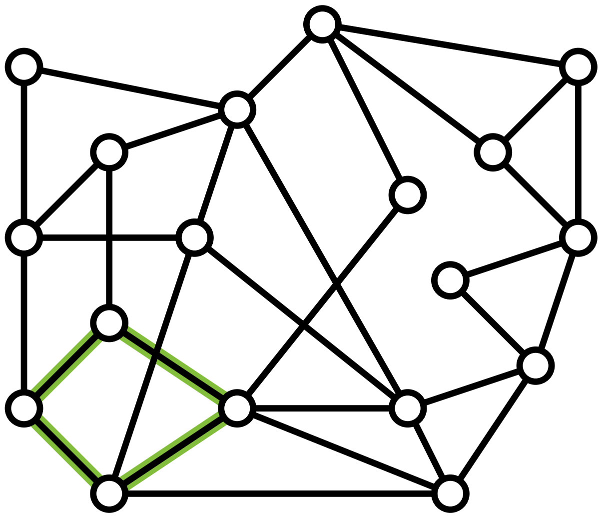
Motivation



- Ein Graph kann sehr viele Kreise haben.

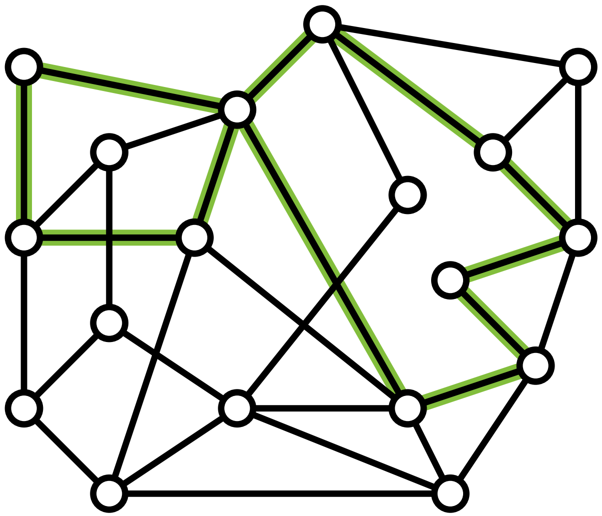
Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation



- Ein Graph kann sehr viele Kreise haben.

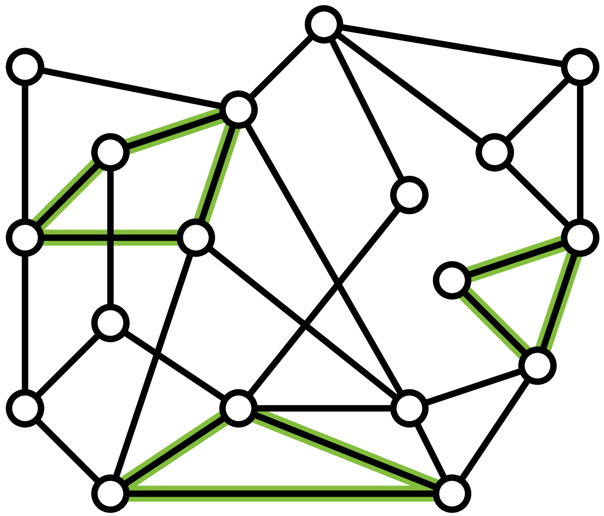
Motivation



- Ein Graph kann sehr viele Kreise haben.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

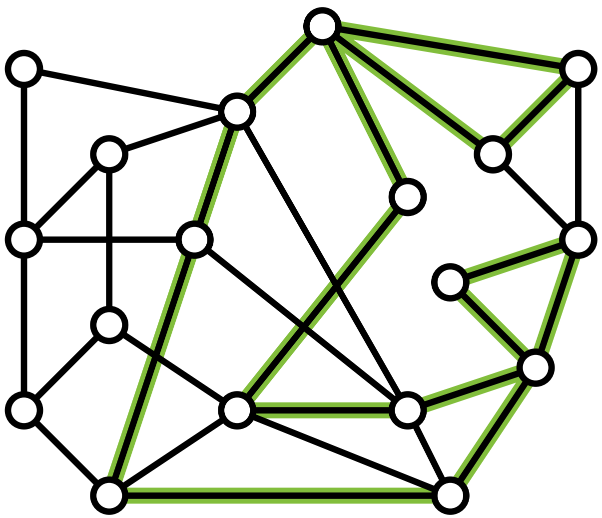
Motivation



- Ein Graph kann sehr viele Kreise haben.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

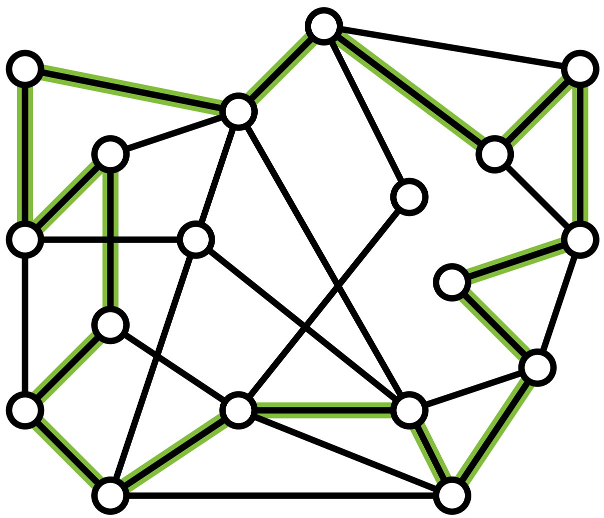
Motivation



- Ein Graph kann sehr viele Kreise haben.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

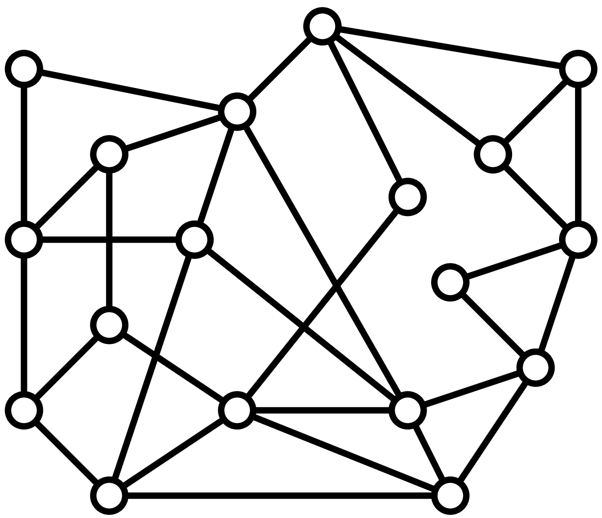
Motivation



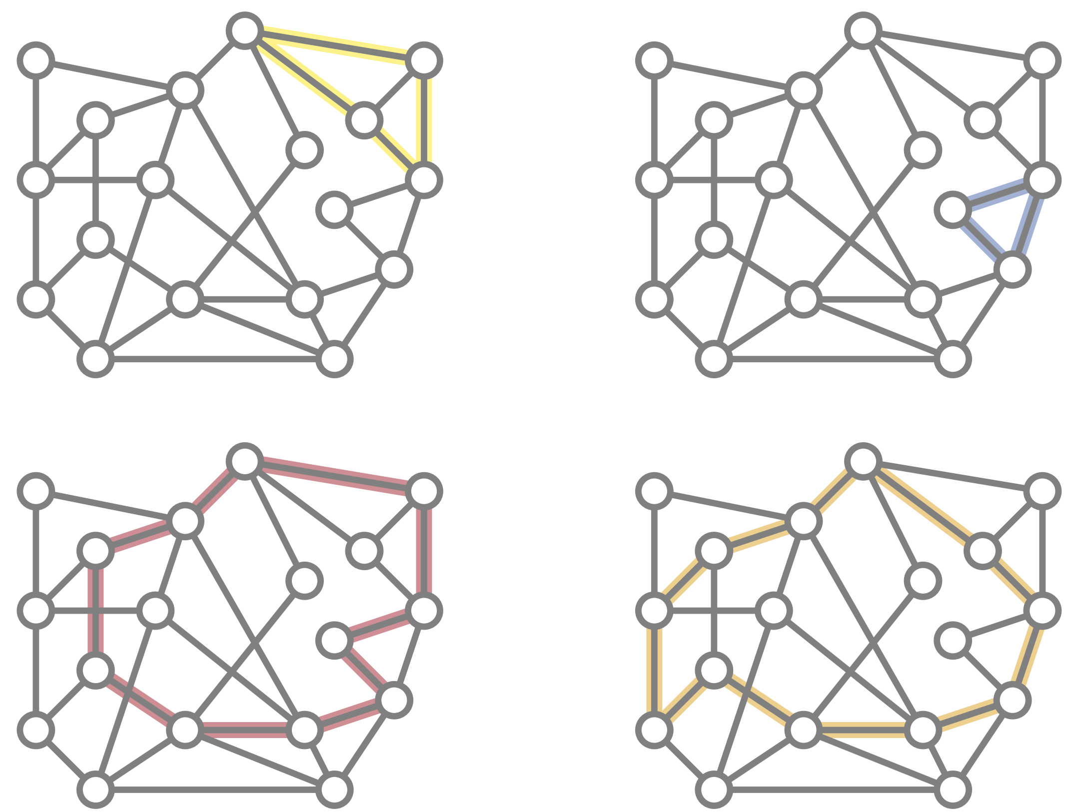
- Ein Graph kann sehr viele Kreise haben.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation

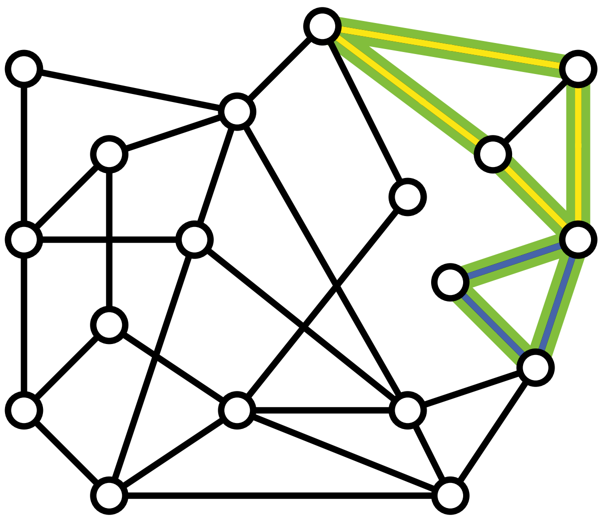


- Ein Graph kann sehr viele Kreise haben.

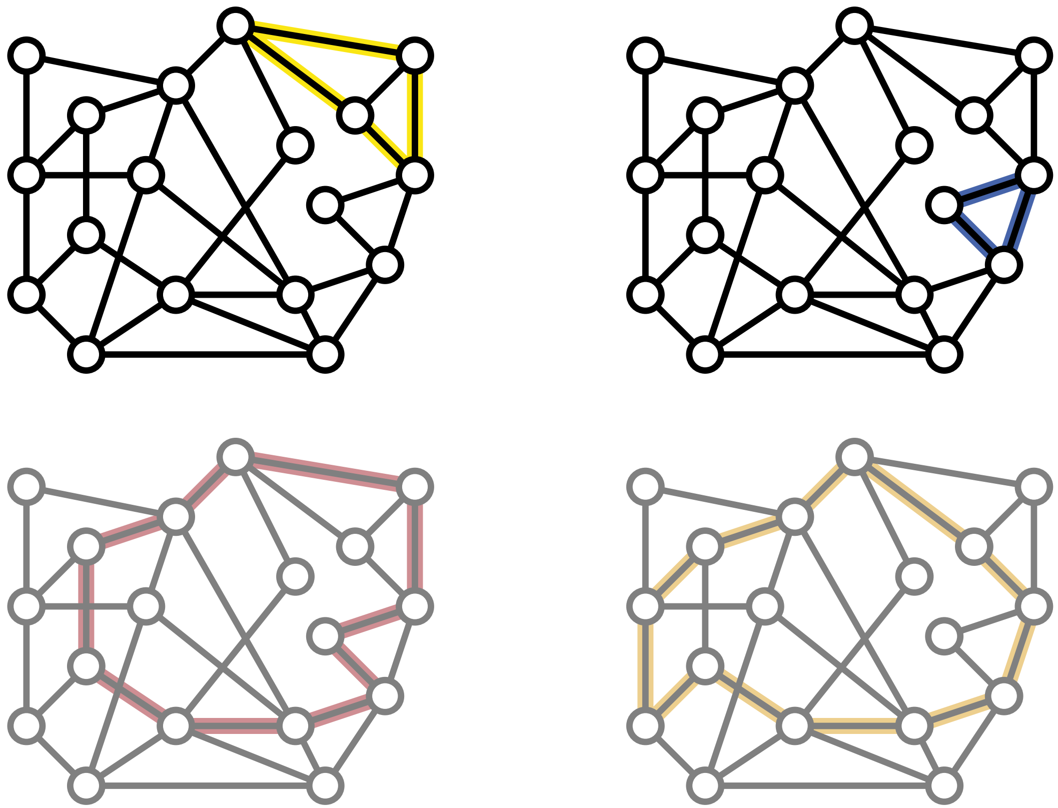


- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Motivation



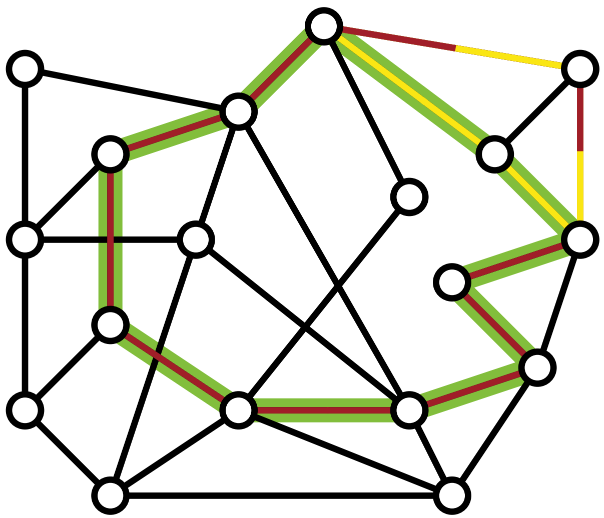
- Ein Graph kann sehr viele Kreise haben.



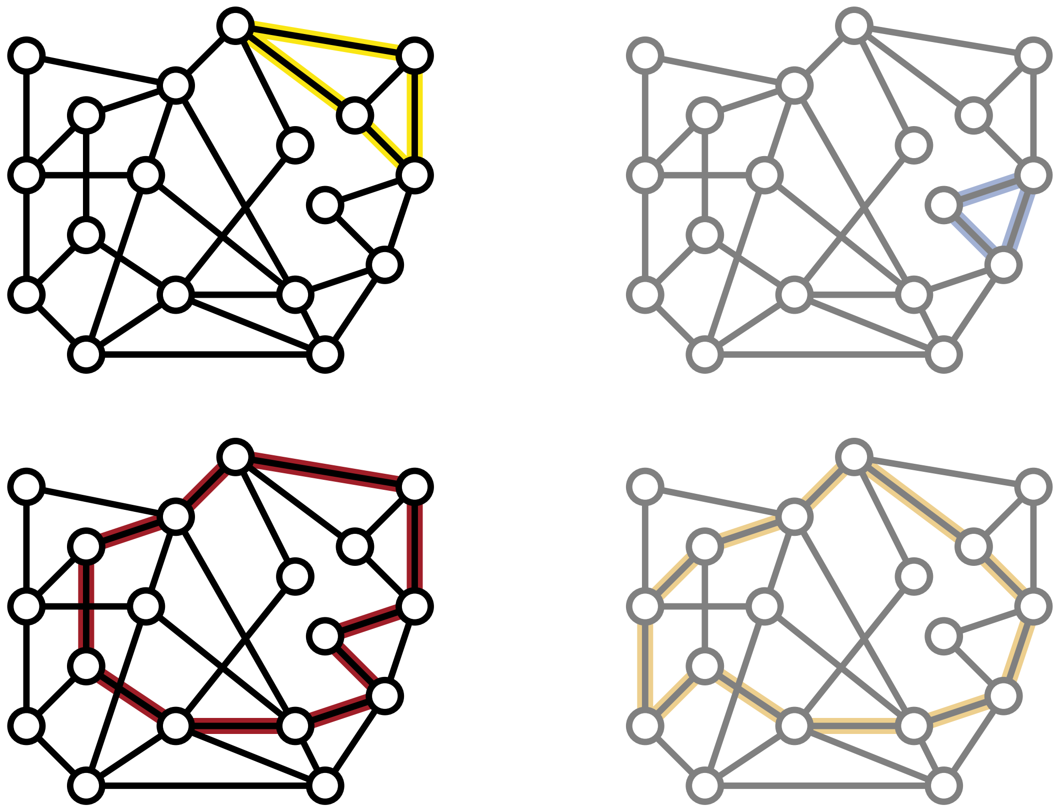
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation

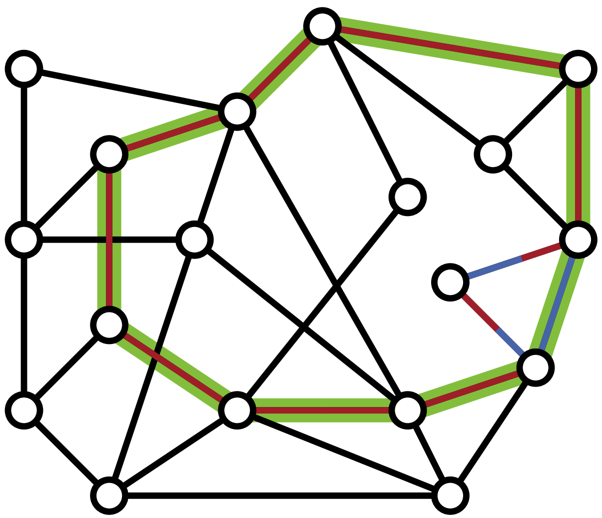


- Ein Graph kann sehr viele Kreise haben.

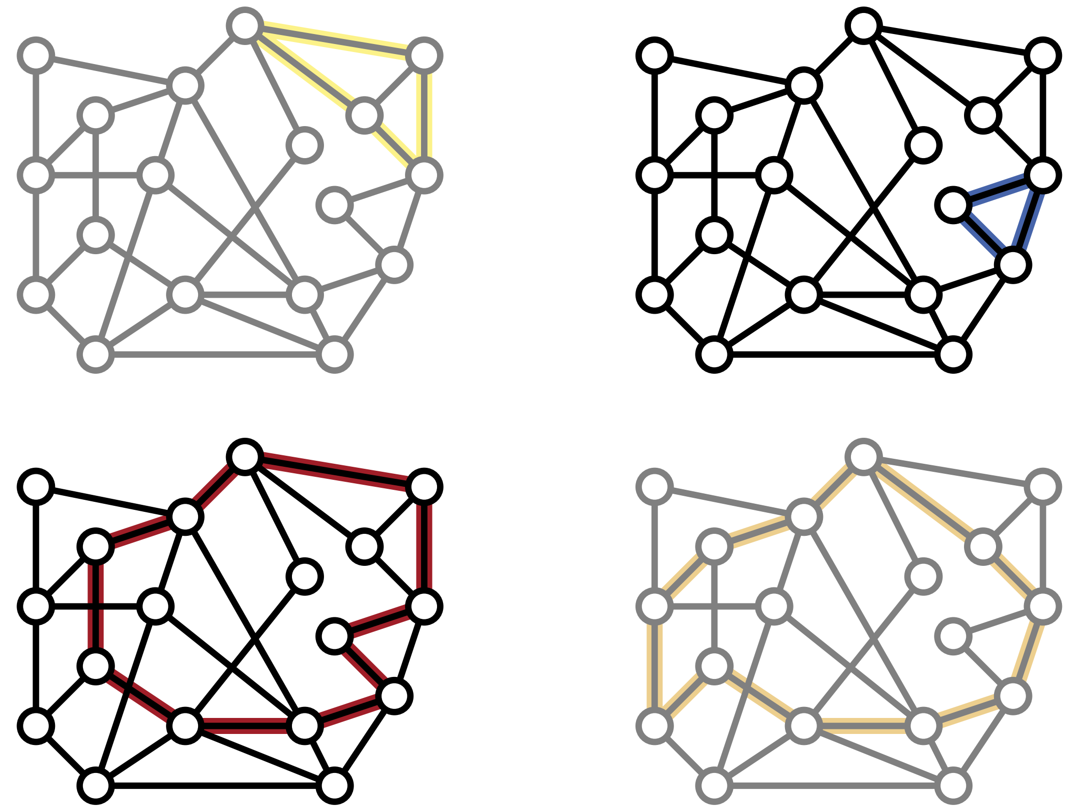


- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Motivation

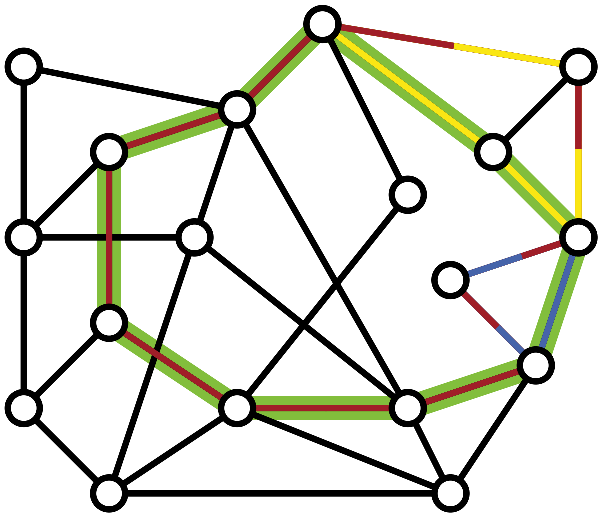


- Ein Graph kann sehr viele Kreise haben.

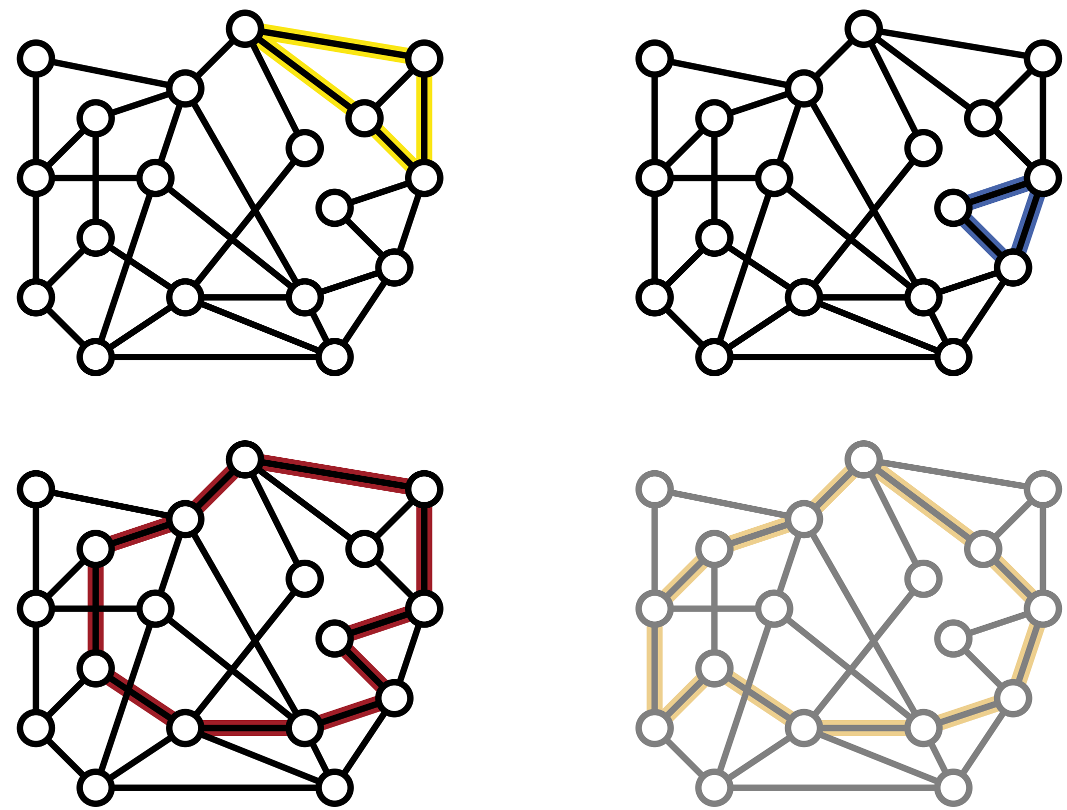


- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Motivation



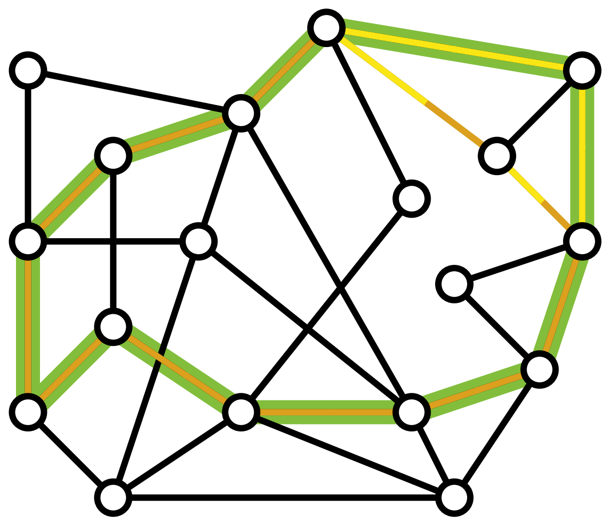
- Ein Graph kann sehr viele Kreise haben.



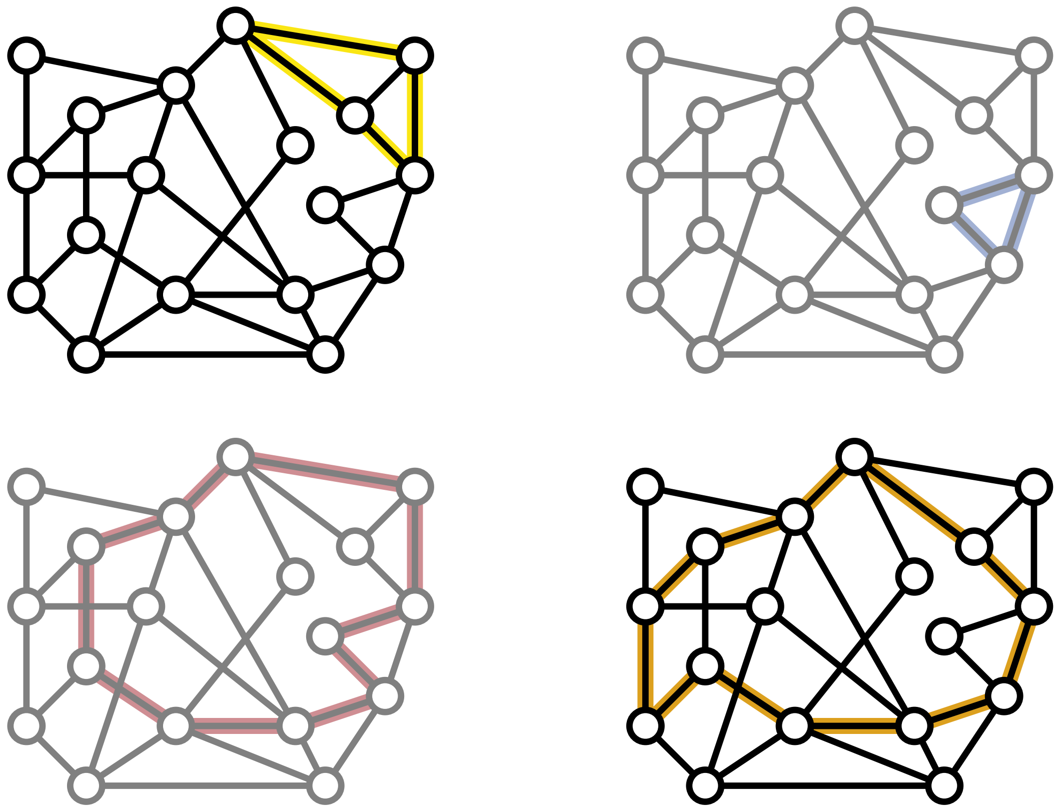
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation



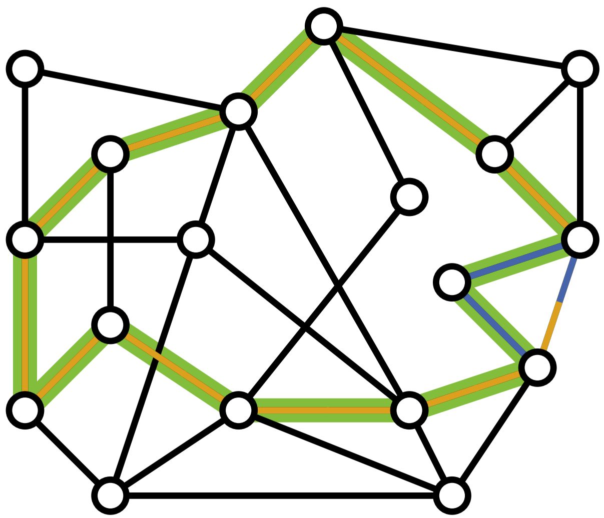
- Ein Graph kann sehr viele Kreise haben.



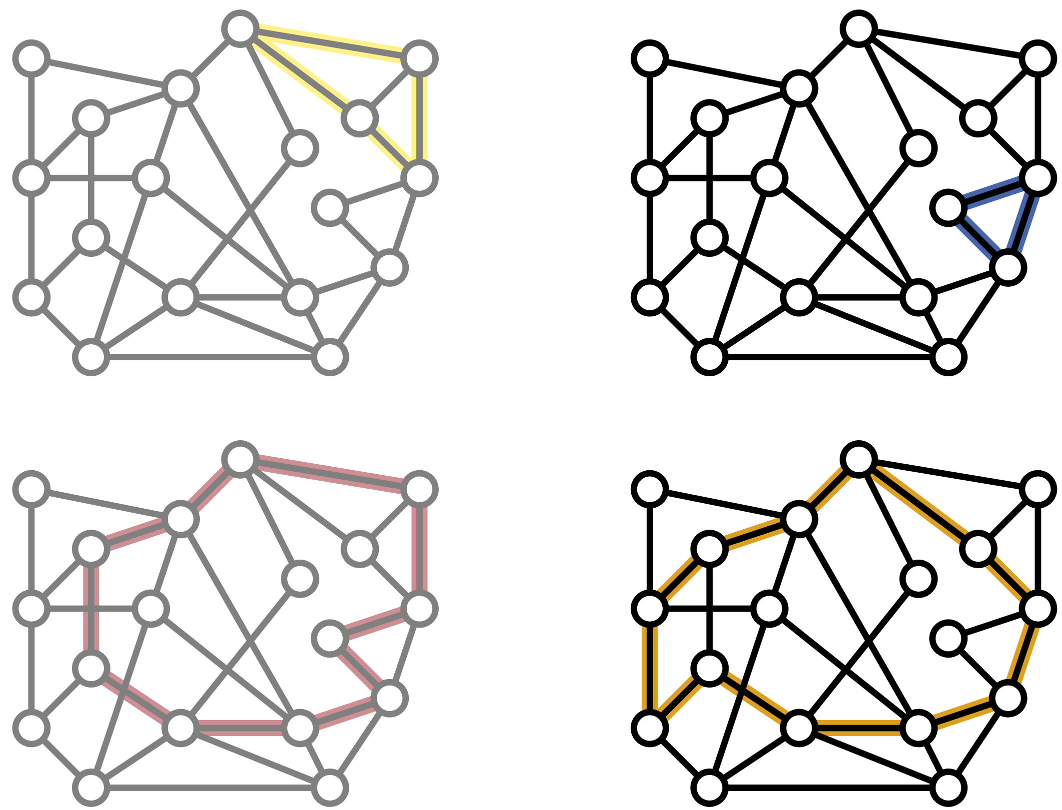
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation

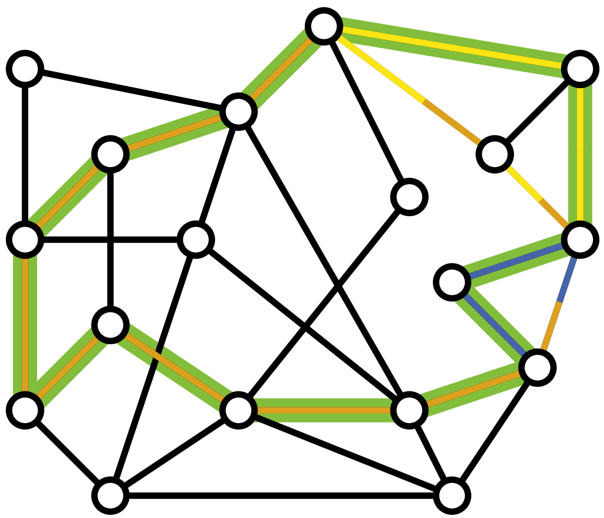


- Ein Graph kann sehr viele Kreise haben.

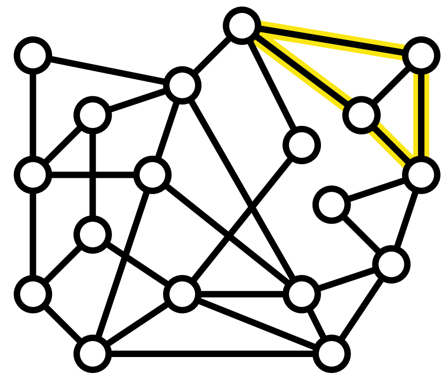


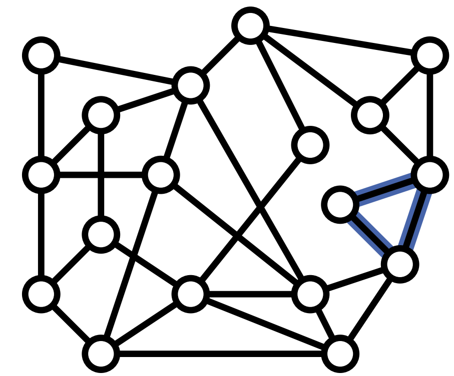
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

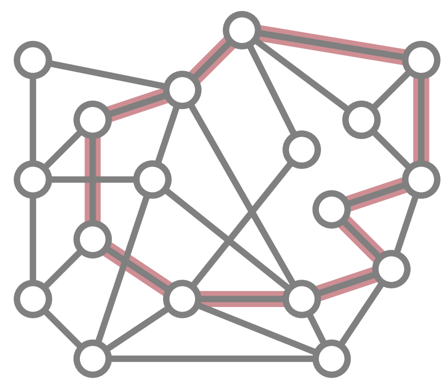
Motivation

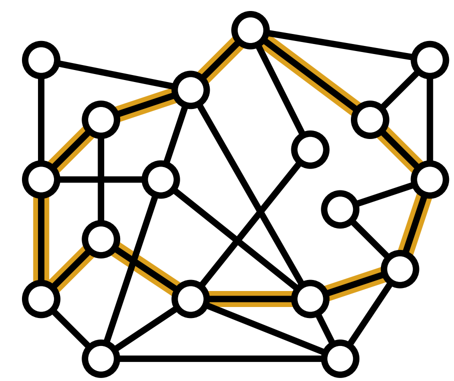


- Ein Graph kann sehr viele Kreise haben.



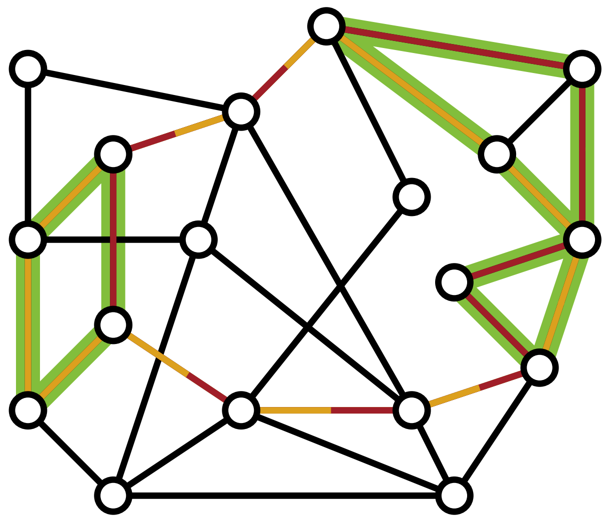




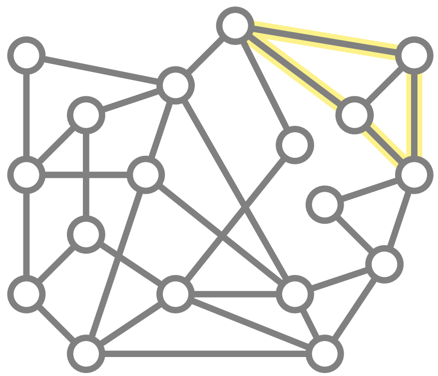


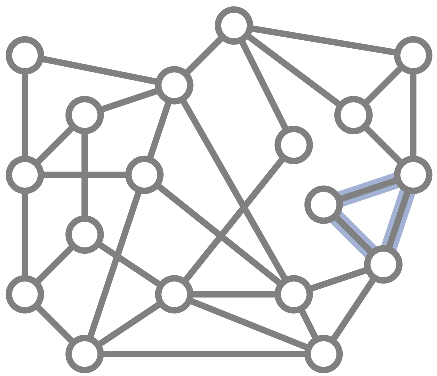
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Motivation

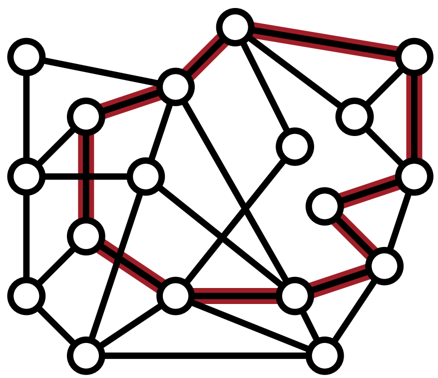


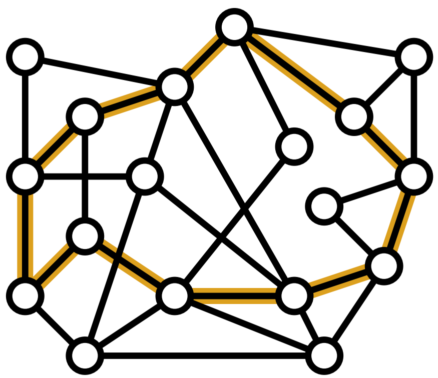
- Ein Graph kann sehr viele Kreise haben.





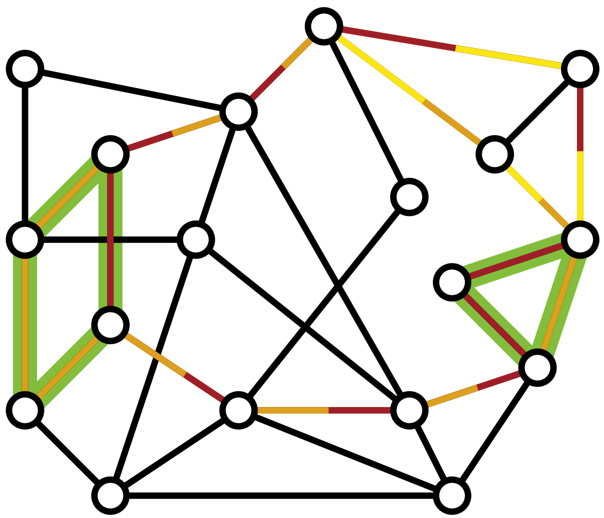
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?



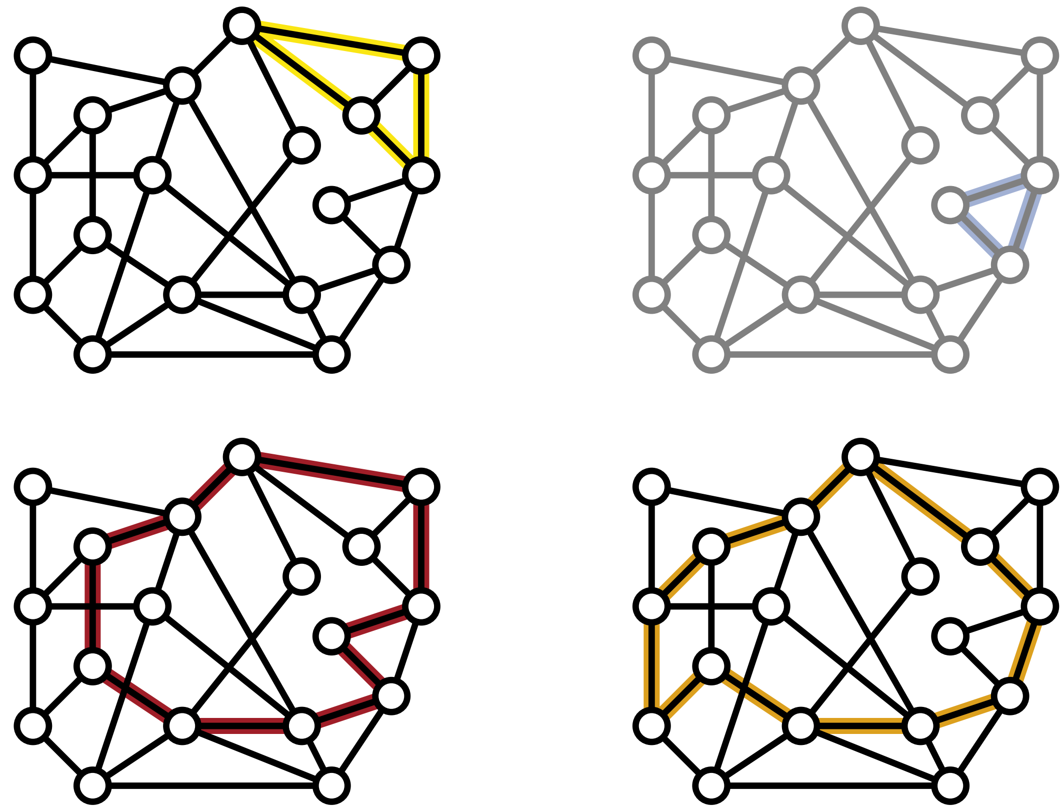


Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation



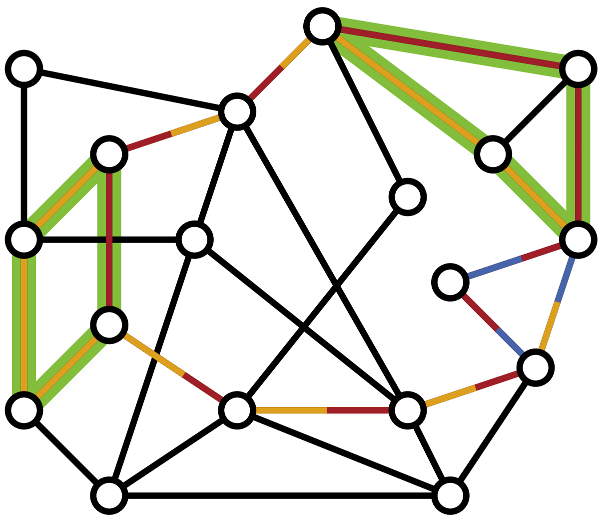
- Ein Graph kann sehr viele Kreise haben.



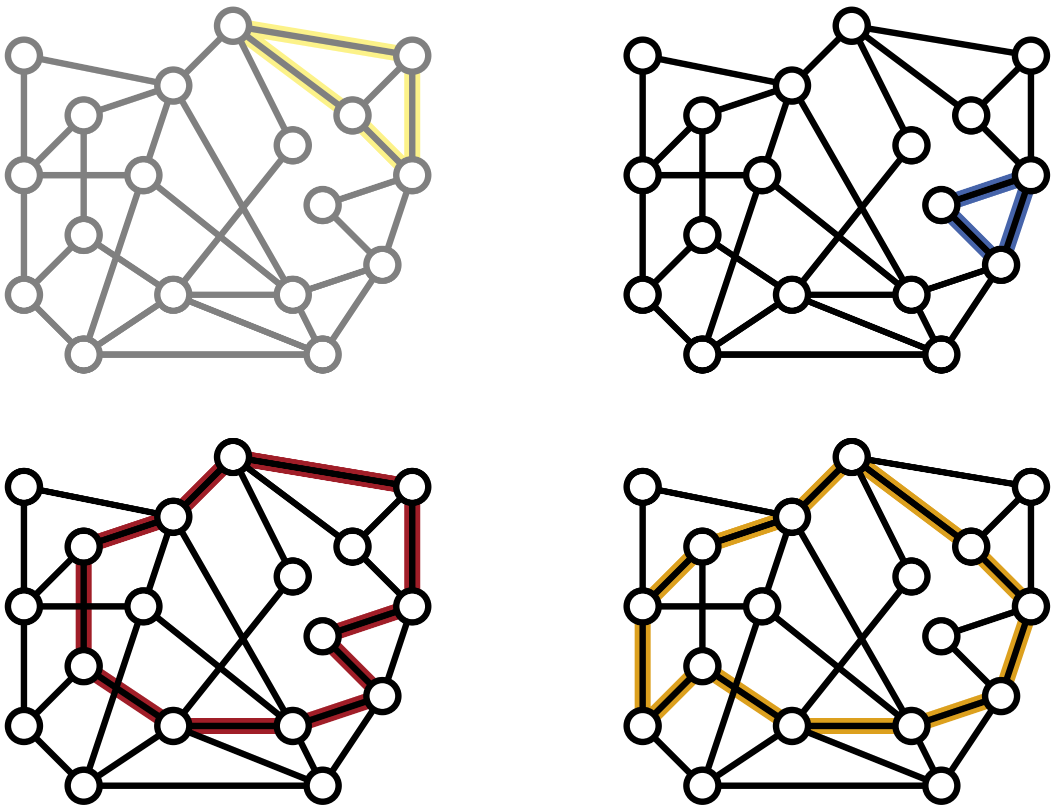
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation



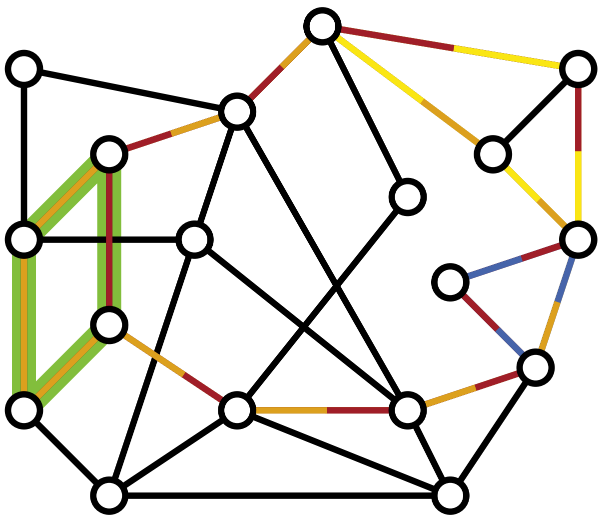
- Ein Graph kann sehr viele Kreise haben.



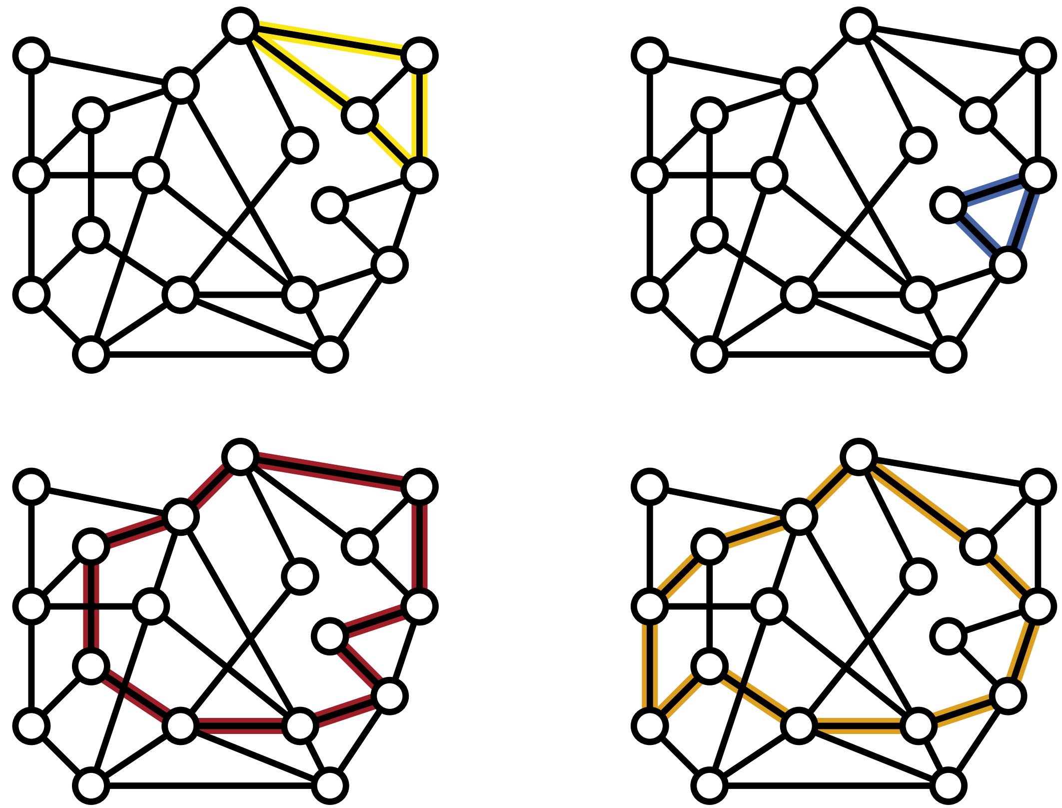
- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Motivation



- Ein Graph kann sehr viele Kreise haben.

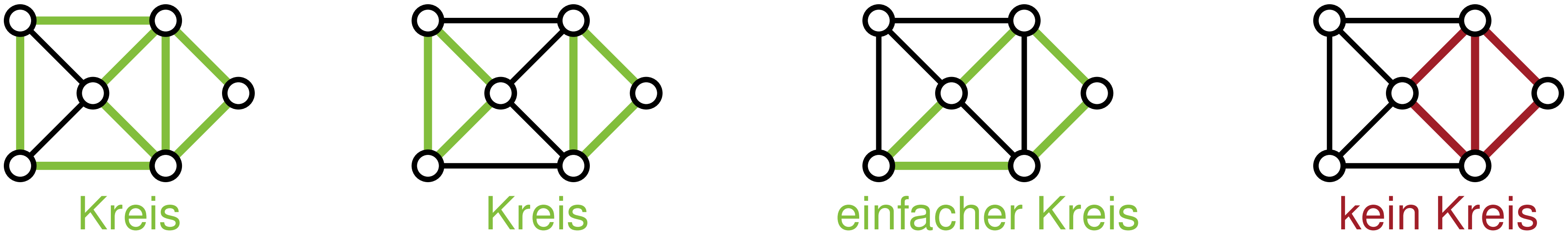


- Man kann aus wenigen Kreisen viele zusammensetzen.
- Wie viele braucht man, um alle Kreise zu erzeugen?

Kreise in Graphen (Wiederholung)

Definition: Kreis **(Definition 5.1)**

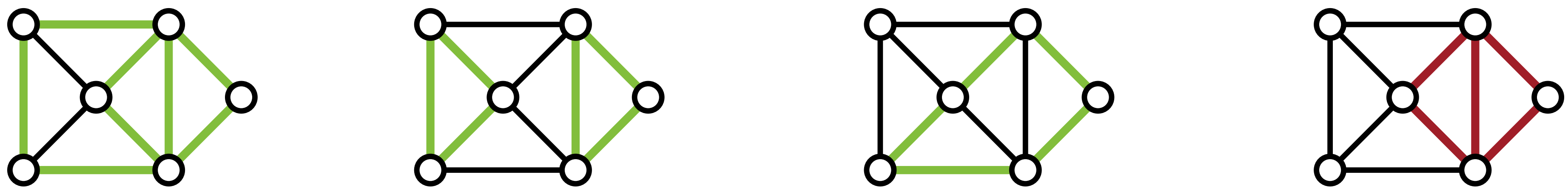
Ein Teilgraph $C = (V_C, E_C)$ von $G = (V, E)$ (d.h. $V_c \subseteq V$, $E_c \subseteq E$) heißt *Kreis* in G, falls alle Knoten aus V_C in C geraden Grad haben. Falls C zusammenhängend ist und alle Knoten aus V_C Grad zwei haben, so heißt C *einfacher Kreis*.



Kreise in Graphen (Wiederholung)

Definition: Kreis **(Definition 5.1)**

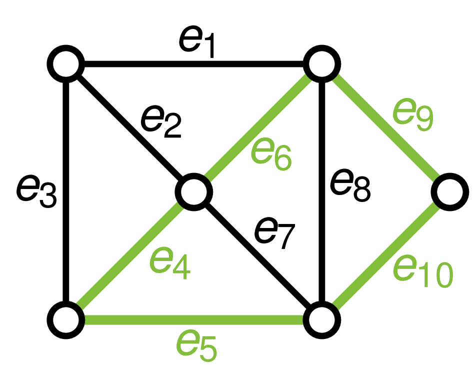
Ein Teilgraph $C = (V_C, E_C)$ von $G = (V, E)$ (d.h. $V_c \subseteq V$, $E_c \subseteq E$) heißt *Kreis* in G, falls alle Knoten aus V_C in C geraden Grad haben. Falls C zusammenhängend ist und alle Knoten aus V_C Grad zwei haben, so heißt C *einfacher Kreis*.



Kreis — Kreis — einfacher Kreis — kein Kreis

Fasse Kreis als Kantenmenge $E' \subseteq E = \{e_1, \ldots, e_m\}$ auf und kodiere E' als Vektor $X^{E'}$ mit

$$X_i^{E'} := \begin{cases} 1, & \text{falls } e_i \in E' \\ 0, & \text{sonst} \end{cases}$$



$$\longrightarrow \quad X^{E'} = \begin{array}{cc} \left(\begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{array}\right) & \begin{array}{c} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \\ e_{10} \end{array} \end{array}$$

Kreisraum (Wiederholung)

Definition: Kreisraum
Sei $\mathcal{C}$ die Menge aller Kreise in $G = (V, E)$. Dann induziert $\mathcal{C}$ den Vektorraum der Vektoren X^c, $c \in \mathcal{C}$ über dem Körper GF(2), genannt *Kreisraum* von G.

Erinnerung: $GF(2)$ ist der Körper mit zwei Elementen $\{0, 1\}$ und den Verknüpfungen + und $\cdot$ mit

+	0	1
0	0	1
1	1	0

$\cdot$	0	1
0	0	0
1	0	1

Kreisraum (Wiederholung)

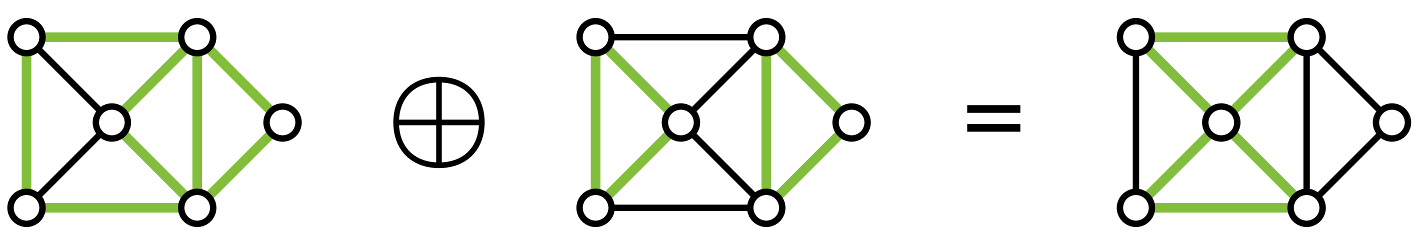
Definition: Kreisraum
Sei $\mathcal{C}$ die Menge aller Kreise in $G = (V, E)$. Dann induziert $\mathcal{C}$ den Vektorraum der Vektoren X^c, $c \in \mathcal{C}$ über dem Körper GF(2), genannt *Kreisraum* von G.

Erinnerung: $GF(2)$ ist der Körper mit zwei Elementen $\{0, 1\}$ und den Verknüpfungen $+$ und $\cdot$ mit

$+$	0	1
0	0	1
1	1	0

$\cdot$	0	1
0	0	0
1	0	1

Definition: Summe von Kreisen – symmetrische Differenz
Die Addition im Kreisraum von G induziert eine Operation $\oplus$ auf C durch $c_1 \oplus c_2 = (E_{c_1} \cup E_{c_2}) \setminus (E_{c_1} \cap E_{c_2})$. Dies ist die *symmetrische Differenz* beider Kantenmengen.



Ist wieder ein Kreis!

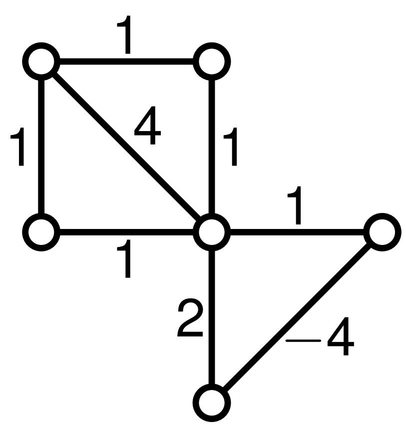
Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Problem – Minimum Cycle Basis

Definition: Gewicht einer Kreisbasis
Sei zu $G = (V, E)$ die Kantengewichtsfunktion $w\colon E \longrightarrow \mathbb{R}_0$ gegeben. Das *Gewicht einer Kreisbasis* $\mathcal{B}$ von G ist definiert als

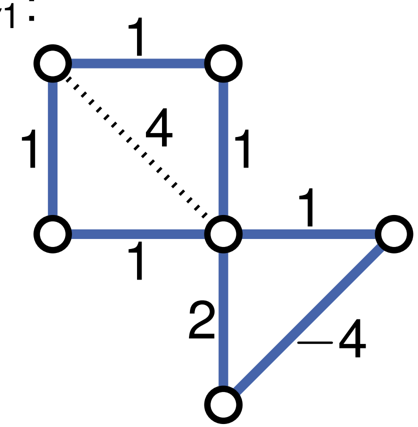
$$w(\mathcal{B}) = \sum_{C \in \mathcal{B}} w(C) = \sum_{C \in \mathcal{B}} \sum_{e \in C} w(e)$$

Problem: MCB
Gegeben sei ein Graph $G = (V, E)$ und eine Gewichtsfunktion $w\colon E \longrightarrow \mathbb{R}_0$. Finde eine Kreisbasis $\mathcal{B}$ von G mit minimalem Gewicht.



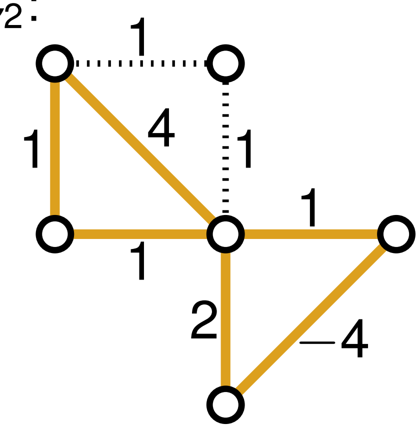
G=(V,E)

C_1:



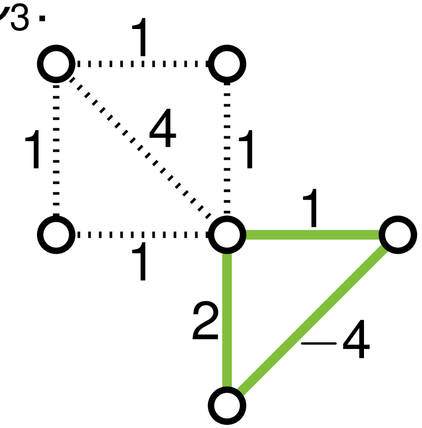
$w(C_1) = 3$

C_2:



$w(C_2) = 5$

C_3:



$w(C_3) = -1$

Weshalb $\{C_1, C_2, C_3\}$ MCB ist, auf folgenden Folien.

Kreismatroid

Karlsruher Institut für Technologie

Sei $\mathcal{C}$ die Menge aller Kreise in $G = (V, E)$ und $\mathcal{U}$ die Menge aller unabhängigen Teilmengen von $\mathcal{C}$.
Dann bildet $(\mathcal{C}, \mathcal{U})$ ein **Unabhängigkeitssystem**.

1. $\emptyset \in U$
2. $I_1 \in \mathcal{U}$, $I_2 \subseteq I_1 \Rightarrow I_2 \in \mathcal{U}$

Definition: Matroid
Ein Unabhängigkeitssystem $(M, \mathcal{U})$ heißt *Matroid*, wenn für alle $I, J \in \mathcal{U}$ mit $|I| < |J|$, ein $e \in J \setminus I$ existiert, sodass $I \cup \{e\} \in \mathcal{U}$.

Satz 5.6: Das Unabhängigkeitssystem $(\mathcal{C}, \mathcal{U})$ ist ein Matroid.

Kreismatroid

Beweis: Aus dem Austausschsatz von Steinitz folgt dies für jeden Vektorraum (Übung).

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Greedy-Lösung

In letzter Vorlesung gezeigt: Greedy-Methode optimal für Optimierungsprobleme auf Matroiden.

GREEDY-METHODE für MCB

Eingabe: Menge $\mathcal{C}$ aller Kreise in $G = (V, E)$.
Ausgabe: Kreisbasis minimalen Gewichts von G.

Sortiere $\mathcal{C}$ aufsteigend nach Gewicht zu $C_1, \ldots, C_k$.

$\mathcal{B}^\star \leftarrow \emptyset$

for $i = 1$ **to** k **do**
  if $\mathcal{B}^\star \cup \{C_i\}$ *linear unabhängig* **then**
    $B^\star \leftarrow B^\star \cup \{C_i\}$

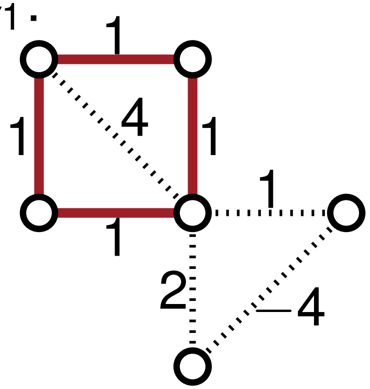
Greedy-Methode liefert also MCB, da

Satz 5.6: Das Unabhängigkeitssystem $(\mathcal{C}, \mathcal{U})$ ist ein Matroid.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

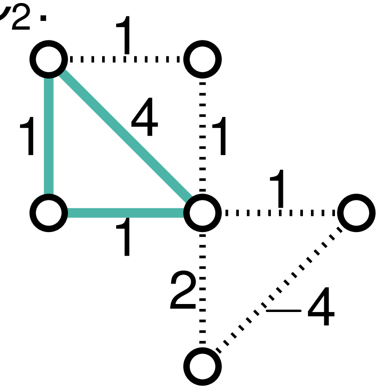
Beispiel

C_1:



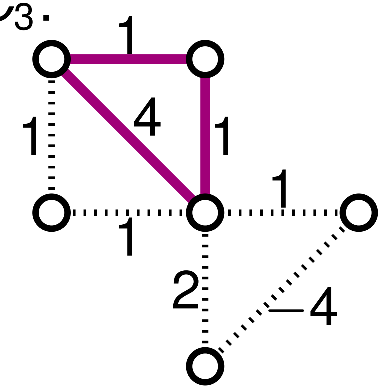
$w(C_1) = 4$

C_2:



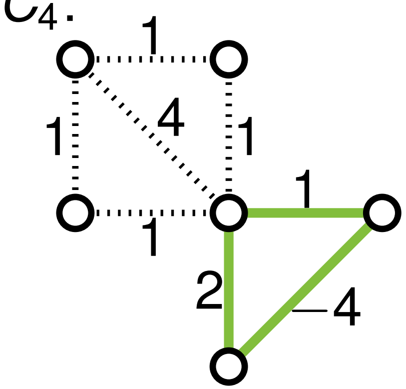
$w(C_2) = 6$

C_3:



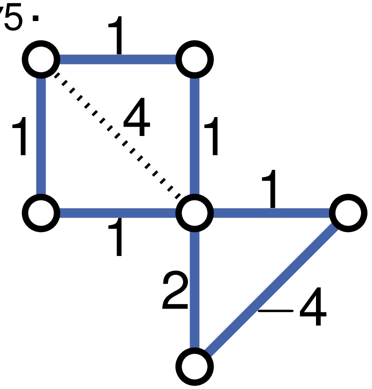
$w(C_3) = 6$

C_4:



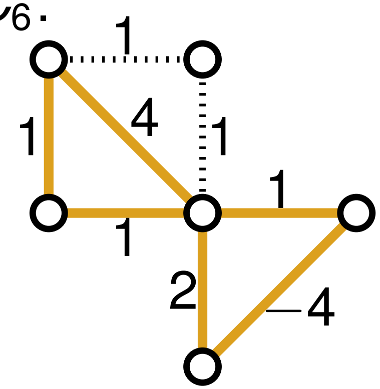
$w(C_4) = -1$

C_5:



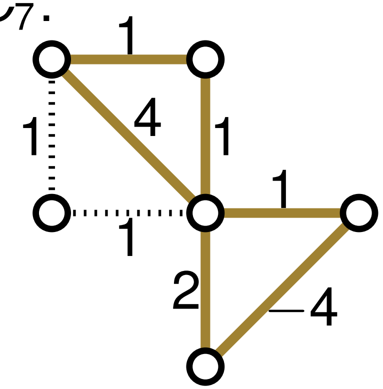
$w(C_5) = 3$

C_6:



$w(C_6) = 5$

C_7:



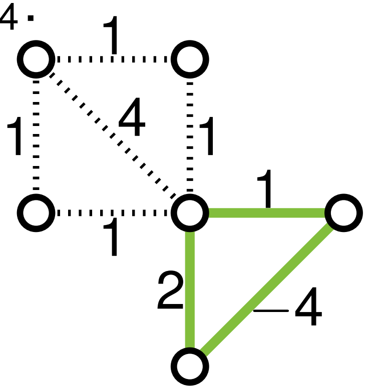
$w(C_7) = 5$

1. Schritt: Zähle alle Kreise auf.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

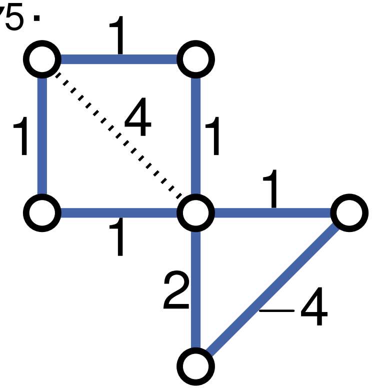
Beispiel

C_4:



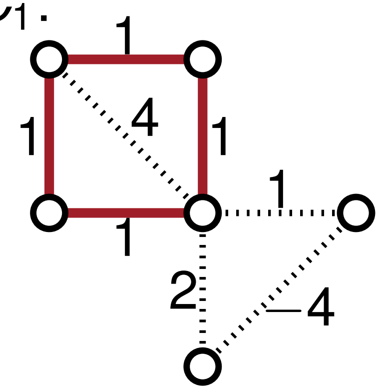
$w(C_4) = -1$

C_5:



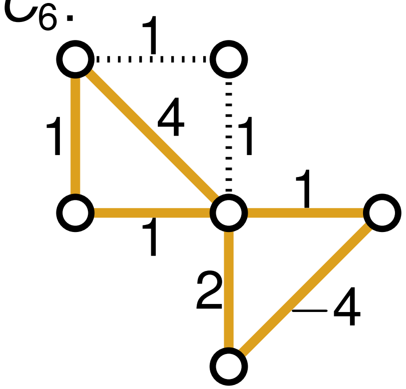
$w(C_5) = 3$

C_1:



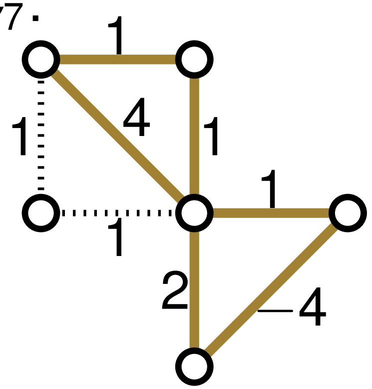
$w(C_1) = 4$

C_6:



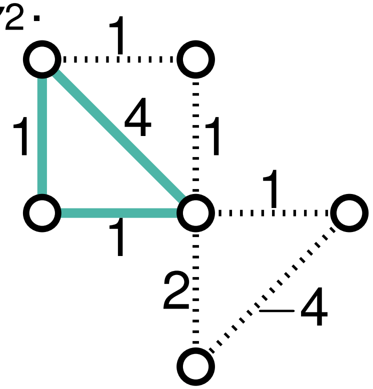
$w(C_6) = 5$

C_7:



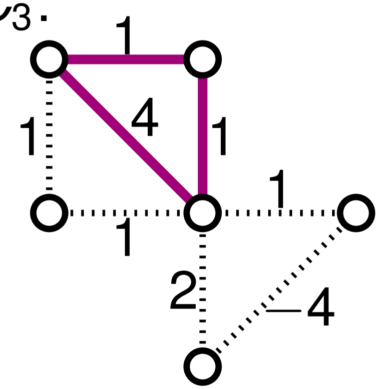
$w(C_7) = 5$

C_2:



$w(C_2) = 6$

C_3:

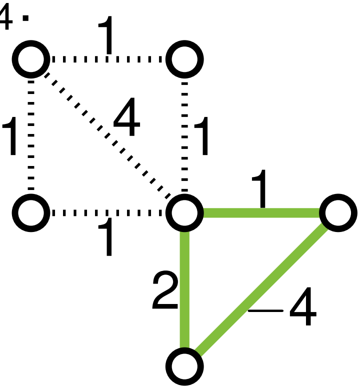


$w(C_3) = 6$

2. Schritt: Sortiere Kreise nach Gewicht.

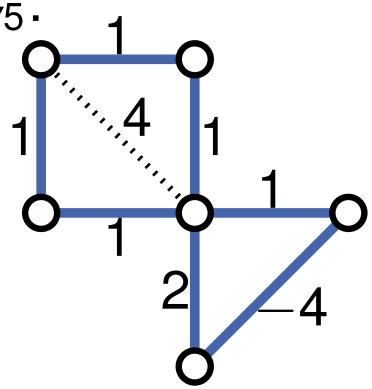
Beispiel

C_4:



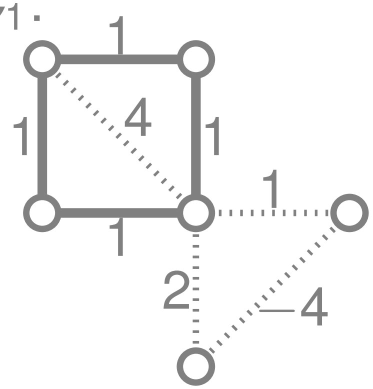
$w(C_4) = -1$

C_5:



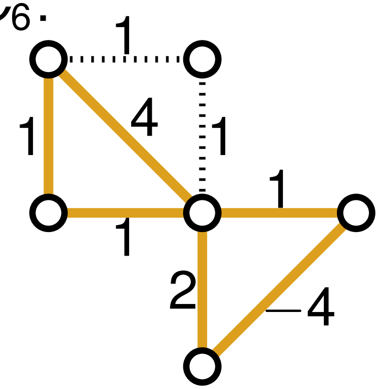
$w(C_5) = 3$

C_1:



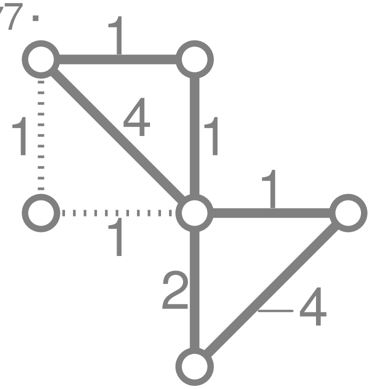
$w(C_1) = 4$

C_6:



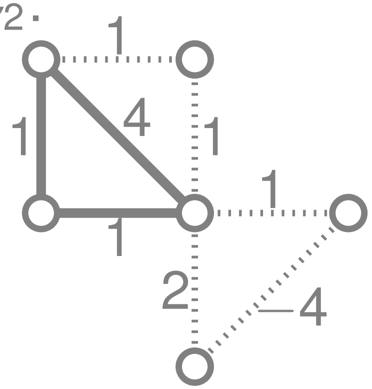
$w(C_6) = 5$

C_7:



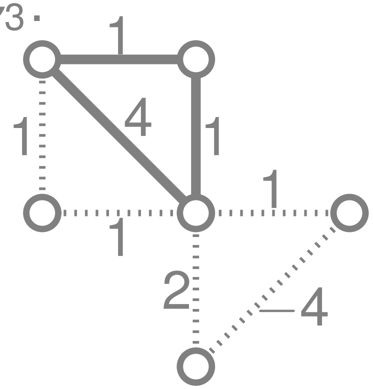
$w(C_7) = 5$

C_2:



$w(C_2) = 6$

C_3:



$w(C_3) = 6$

3. Schritt: Bestimme maximale unabhängige Menge nach Greedy-Prinzip.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Greedy-Lösung

In letzter Vorlesung gezeigt: Greedy-Methode optimal für Optimierungsprobleme auf Matroiden.

GREEDY-METHODE für MCB

Eingabe: Menge $\mathcal{C}$ aller Kreise in $G = (V, E)$.
Ausgabe: Kreisbasis minimalen Gewichts von G.

Sortiere $\mathcal{C}$ aufsteigend nach Gewicht zu $C_1, \dots, C_k$.

$\mathcal{B}^\star \leftarrow \emptyset$

for $i = 1$ **to** k **do**
  if $\mathcal{B}^\star \cup \{C_i\}$ *linear unabhängig* **then**
    $B^\star \leftarrow B^\star \cup \{C_i\}$

Problem: Anzahl Kreise in einem Graphen im Allgemeinen exponentiell in Anzahl Kanten und Knoten des Graphen (siehe Übung).

Idee: Versuche Kandidatenmenge mit polynomieller Größe zu finden.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Annahmen und Beobachtungen

Ab jetzt: Positive Gewichtsfunktion, also $w\colon E \to \mathbb{R}^+$

→ Jeder Kreis einer MCB ist einfach.

G ist zusammenhängender Graph.

→ Keine wirkliche Einschränkung, da Komponenten unabhängig betrachtet werden können.

Weitere Beobachtung:

Für jede Kante e in G, die in einem Kreis enthalten ist, gilt:

In jeder Kreisbasis gibt es einen Kreis, der e enthält.

Nenne Kreis mit minimalen Gewicht, der e enthält, *kürzesten Kreis* der e enthält.

Algorithmus von Horton

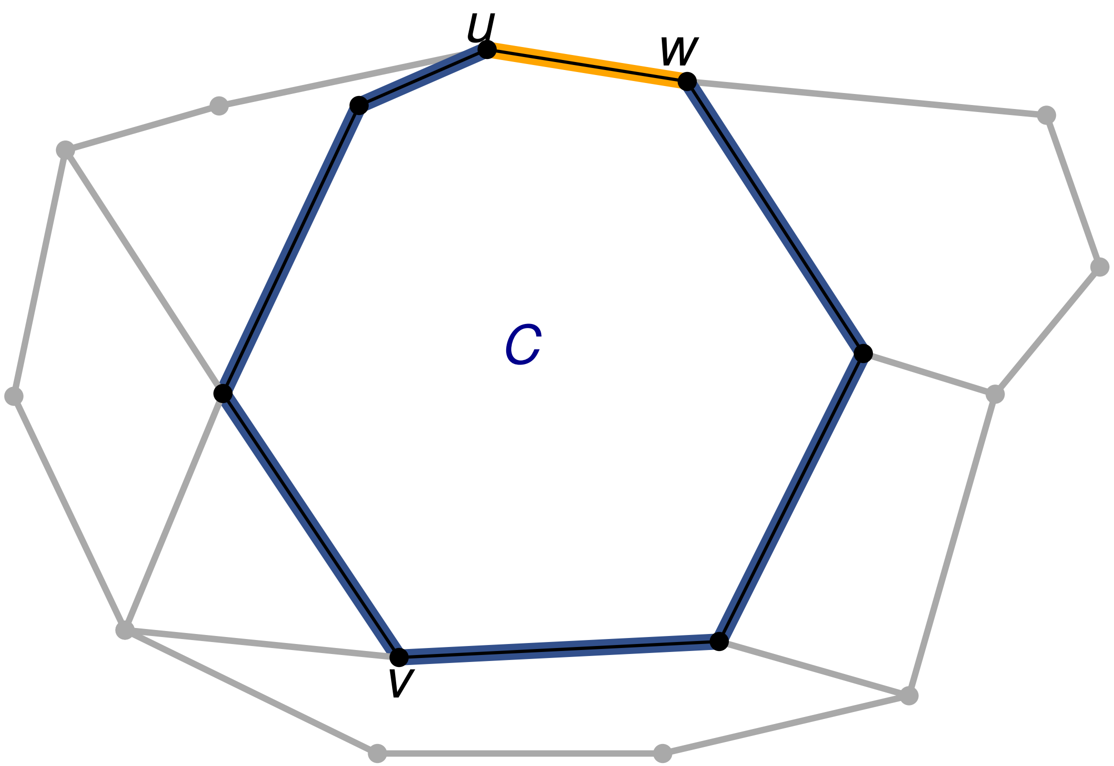
Algorithmus von Horton

Satz 5.7 Für jeden Kreis C aus einer MCB von G existiert zu jedem beliebigen Knoten v aus C eine Kante $\{u, w\}$ auf C, sodass gilt:

$$C = \text{SP}(u, v) + \text{SP}(w, v) + \{u, w\},$$

wobei $\text{SP}(u, v)$ bzw. $\text{SP}(w, v)$ ein kürzester Weg von u bzw. w nach v in G ist.

Beweis mithilfe der folgenden Lemmata.



Algorithmus von Horton

Lemma 5.8
Falls $\mathcal{B}$ eine Kreisbasis ist, $C \in \mathcal{B}$ und $C = C_1 \oplus C_2$, dann ist entweder $\mathcal{B} \setminus \{C\} \cup \{C_1\}$ oder $\mathcal{B} \setminus \{C\} \cup \{C_2\}$ wieder eine Kreisbasis.

Algorithmus von Horton

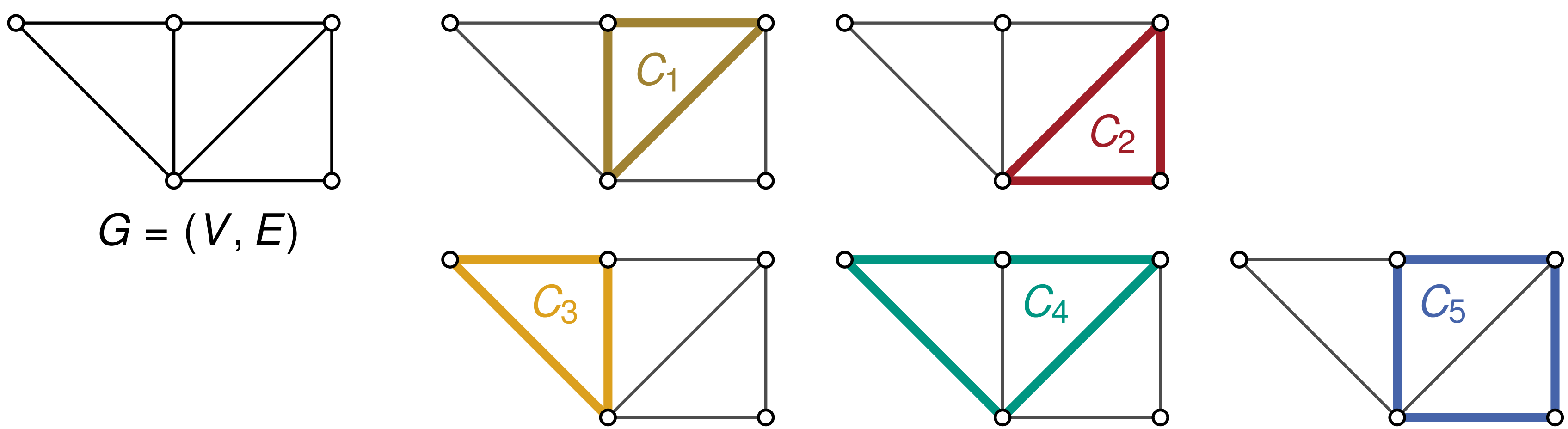
Lemma 5.8
Falls $\mathcal{B}$ eine Kreisbasis ist, $C \in \mathcal{B}$ und $C = C_1 \oplus C_2$, dann ist entweder $\mathcal{B} \setminus \{C\} \cup \{C_1\}$ oder $\mathcal{B} \setminus \{C\} \cup \{C_2\}$ wieder eine Kreisbasis.

Beweis:

1. Fall: C_1 darstellbar als Linerarkombination von Kreisen aus $\mathcal{B} \setminus \{C\}$

$\longrightarrow$ $\mathcal{B} \setminus \{C\} \cup \{C_2\}$ ist Basis.

Beispiel:



$\mathcal{B} = \{C_3, C_4, C_5\}$ ist Basis.

Es gilt: $C_5 = C_1 \oplus C_2$ und $C_1 = C_3 \oplus C_4$

$\longrightarrow$ $\mathcal{B} \setminus \{C_5\} \cup \{C_2\}$ ist Basis.

Algorithmus von Horton

Lemma 5.8
Falls $\mathcal{B}$ eine Kreisbasis ist, $C \in \mathcal{B}$ und $C = C_1 \oplus C_2$, dann ist entweder $\mathcal{B} \setminus \{C\} \cup \{C_1\}$ oder $\mathcal{B} \setminus \{C\} \cup \{C_2\}$ wieder eine Kreisbasis.

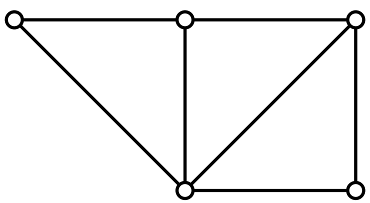
Beweis:

2. Fall: C_1 nur darstellbar als Linearkombination von C und Kreisen aus $\mathcal{B} \setminus \{C\}$

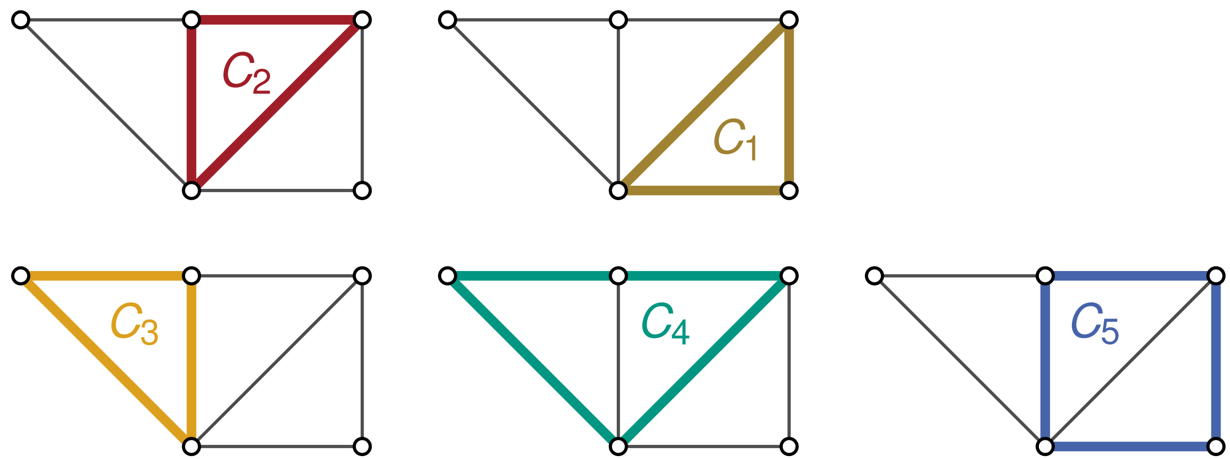
$\longrightarrow$ C_2 darstellbar als Linearkombination von Kreisen aus $\mathcal{B} \setminus \{C\}$

$\longrightarrow$ $\mathcal{B} \setminus \{C\} \cup \{C_1\}$ ist Basis.

Beispiel:



$G = (V, E)$



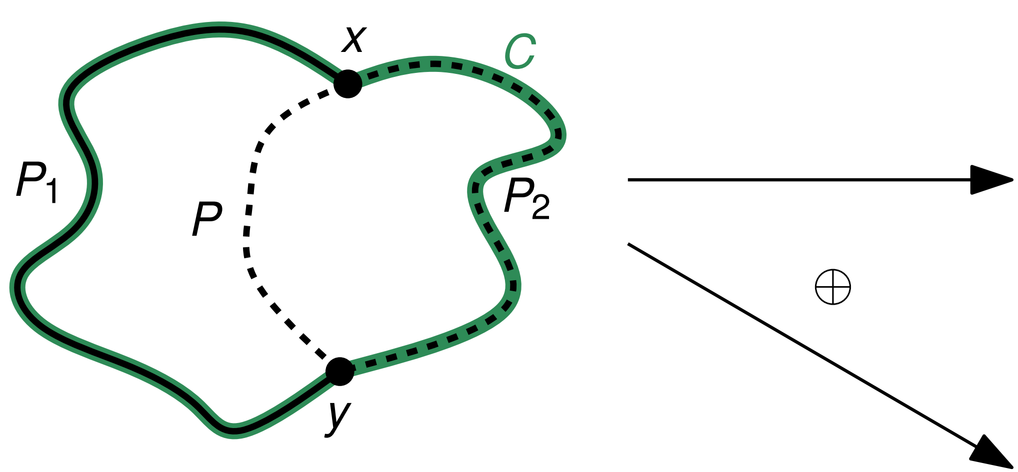
$\left.\begin{array}{l} \mathcal{B} = \{C_3, C_4, C_5\} \text{ ist Basis.} \\ \text{Es gilt: } C_5 = C_1 \oplus C_2 \text{ und } C_1 = \underbrace{C_3 \oplus C_4 \oplus C_5}_{\text{Einzige Kombination.}} \end{array}\right\} \longrightarrow \mathcal{B} \setminus \{C_5\} \cup \{C_1\} \text{ ist Basis.}$

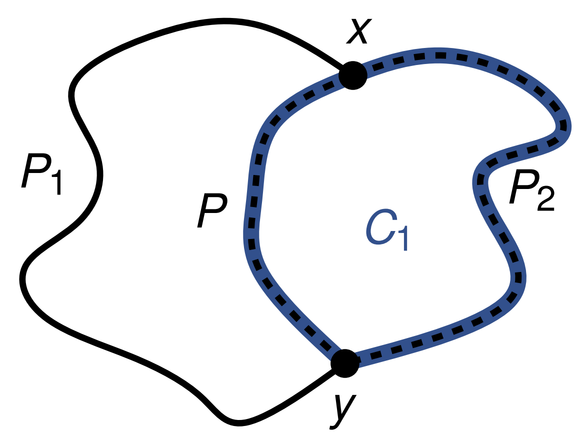
Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

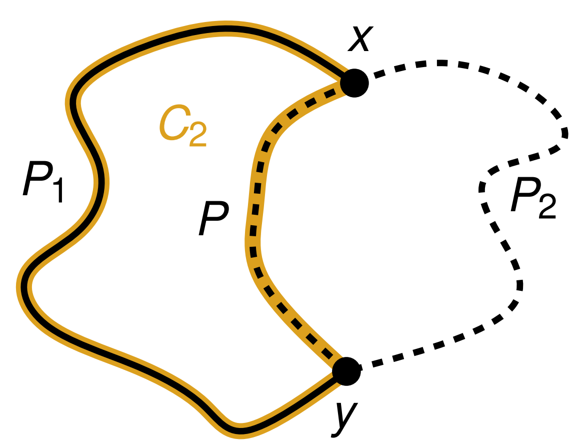
Algorithmus von Horton

Lemma 5.9
Sei $\mathcal{B}$ eine Kreisbasis von G. Für zwei Knoten $x, y \in V$ und einen Weg P in G von x nach y kann jeder Kreis $C \in \mathcal{B}$, der x und y enthält, ersetzt werden durch einen Kreis C', der P enthält.

Beweis:



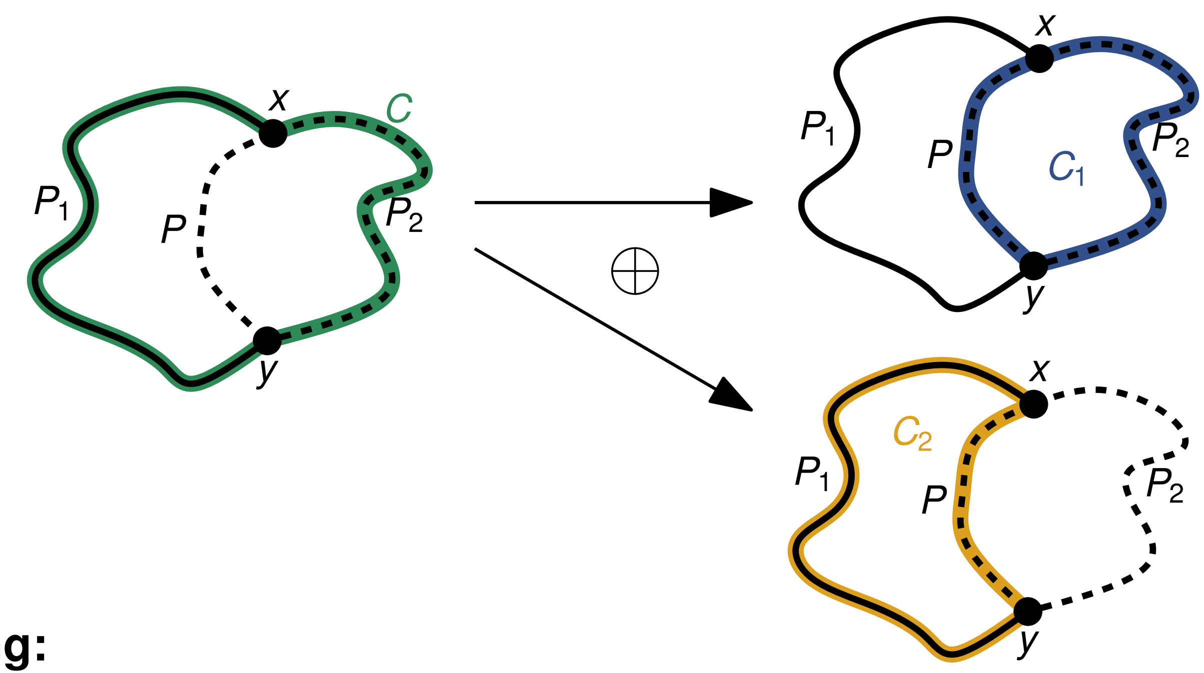




Es gilt $C = C_1 \oplus C_2$ und somit nach Lemma 5.8, dass entweder $\mathcal{B} \setminus \{C\} \cup \{C_1\}$ oder $\mathcal{B} \setminus \{C\} \cup \{C_2\}$ wieder eine Basis ist.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Algorithmus von Horton



Folgerung:
Seien weder P_1 noch P_2 kürzeste Wege zwischen x und y und sei P kürzester Weg zwischen x und y

- $\Rightarrow$ $w(C_1) < w(C)$ und $w(C_2) < w(C)$
- $\Rightarrow$ Jede Basis $\mathcal{B}$, die C enthält, kann in Basis $\mathcal{B}'$ umgewandelt werden, die anstatt C entweder C_1 oder C_2 enthält (Lemma 5.9).
- $\Rightarrow$ $w(\mathcal{B}') < w(\mathcal{B})$

Wenn $\mathcal{B}$ eine MCB ist, dann enthält jeder Kreis in $\mathcal{B}$, der $x, y \in V$ enthält, auch einen kürzesten Weg zwischen x und y.

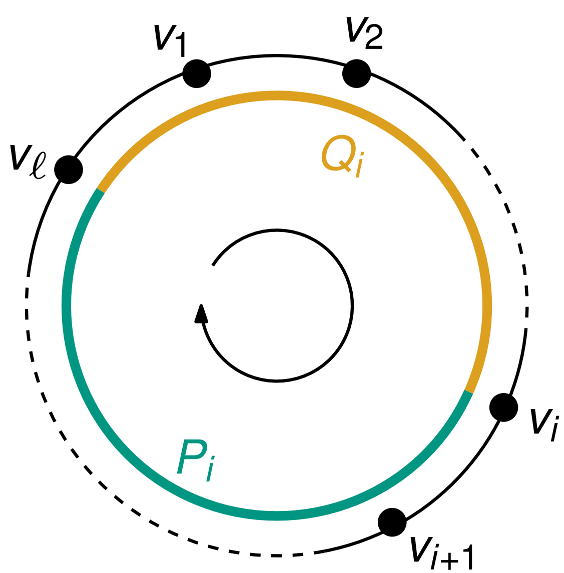
Algorithmus von Horton

Satz 5.7: Für jeden Kreis C aus einer MCB von G existiert zu jedem beliebigen Knoten v aus C eine Kante $\{u, w\}$ auf C, so dass gilt:

$$C = \text{SP}(u, v) + \text{SP}(w, v) + \{u, w\},$$

wobei $\text{SP}(u, v)$ bzw. $\text{SP}(w, v)$ ein kürzester Weg von u bzw. w nach v in G ist.

Beweis:



Betrachte beliebigen Kreis C der MCB, sowie einen belibiegen Knoten v auf C:

- Indizierung der Knoten auf Kreis sei $v = v_0, \ldots, v_\ell = v$.
- Q_i := Weg auf C von v nach v_i in Richtung der Indizierung.
- P_i := Weg auf C von v_i nach v in Richtung der Indizierung.

$\Rightarrow$ Entweder P_i oder Q_i ist kürzester Weg von v nach v_i (vorherige Folie).

- Sei i der größte Index, sodass Q_i kürzester Weg von v nach v_i ist.

$\Rightarrow$ $C = Q_i \oplus \{v_i, v_{i+1}\} \oplus P_{i+1}$ ist gewünschte Darstellung.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Algorithmus von Horton

Algorithmus von Horton für MCB

Eingabe: Zusammenhängender Graph $G = (V, E)$

Ausgabe: Kreisbasis minimalen Gewichts von G.

$\mathcal{H} \leftarrow \emptyset$ — 1. Kandidaten bestimmen

for $v \in V$ *und* $\{u, w\} \in E$ **do**
    Berechne $C_v^{uw} := SP(u, v) + SP(w, v) + \{u, w\}$
    if C_v^{uw} *ist einfach* **then**
        $\mathcal{H} \leftarrow \mathcal{H} \cup \{C_v^{uw}\}$

Sortiere Elemente aus $\mathcal{H}$ aufsteigend zu $C_1, \ldots, C_k$ — 2. Greedy-Prinzip

$\mathcal{B}^\star \leftarrow \emptyset$

for $i = 1$ **to** k **do**
    if $\mathcal{B}^\star \cup \{C_i\}$ *linear unabhängig* **then**
        $B^\star \leftarrow B^\star \cup \{C_i\}$

Algorithmus von Horton

Algorithmus von Horton für MCB

Eingabe: Zusammenhängender Graph $G = (V, E)$

Ausgabe: Kreisbasis minimalen Gewichts von G.

$\mathcal{H} \leftarrow \emptyset$

for $v \in V$ *und* $\{u, w\} \in E$ **do** — $\mathcal{O}(n \cdot m)$, wobei $m \in \mathcal{O}(n^2)$

  Berechne $C_v^{uw} := SP(u, v) + SP(w, v) + \{u, w\}$

  if C_v^{uw} *ist einfach* **then**

    $\mathcal{H} \leftarrow \mathcal{H} \cup \{C_v^{uw}\}$

Sortiere Elemente aus $\mathcal{H}$ aufsteigend zu $C_1, \ldots, C_k$ — $\mathcal{O}(m \cdot n \cdot \log n)$

$\mathcal{B}^\star \leftarrow \emptyset$

for $i = 1$ **to** k **do** — $\mathcal{O}(m^3 \cdot n)$

  if $\mathcal{B}^\star \cup \{C_i\}$ *linear unabhängig* **then** — $\mathcal{O}(m^2)$

    $B^\star \leftarrow B^\star \cup \{C_i\}$

Gesamtlaufzeit: $\mathcal{O}(m^3 \cdot n)$

$\rightarrow |B^\star| \leq m - n + 1$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Algorithmus von Horton

Algorithmus von Horton für MCB

Eingabe: Zusammenhängender Graph $G = (V, E)$

Ausgabe: Kreisbasis minimalen Gewichts von G.

$\mathcal{H} \leftarrow \emptyset$

for $v \in V$ *und* $\{u, w\} \in E$ **do** — $\mathcal{O}(n \cdot m)$, wobei $m \in \mathcal{O}(n^2)$

Sor … og n)

$\mathcal{B}^\star$

Bemerkung:

Algorithmus funktioniert auch für allgemeine Gewichtsfunktionen $E : w \to \mathbb{R}$.

Allerdings:

- Kreise einer MCB nicht mehr unbedingt einfach.
- Finden kürzester Wege ist komplizierter.

for $i = 1$ **to** k **do** — $\mathcal{O}(m^3 \cdot n)$

  if $\mathcal{B}^\star \cup \{C_i\}$ *linear unabhängig* **then** — $\mathcal{O}(m^2)$

    $B^\star \leftarrow B^\star \cup \{C_i\}$

Gesamtlaufzeit: $\mathcal{O}(m^3 \cdot n)$

$|B^\star| \leq m - n + 1$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Algorithmus von de Pina

Algorithmus von de Pina

Sei T ein aufspannender Baum (bzw. Wald) in G und $e_1, \ldots, e_N$ die Nichtbaumkanten aus $G \setminus T$ in einer beliebigen Ordnung, wobei gilt $N = m - n + 1$.

Eingabe: Zusammenhängender Graph $G = (V, E)$

Ausgabe: MCB von G

for $j = 1$ *bis* N **do**
  Initialisiere $S_{1,j} \leftarrow \{e_j\}$

for $k = 1$ *bis* N **do**
  Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält
  for $j = k + 1$ *bis* N **do**

$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Ausgabe ist: $\{C_1, \ldots, C_N\}$

Algorithmus von de Pina

Sei T ein aufspannender Baum (bzw. Wald) in G und $e_1, \ldots, e_N$ die Nichtbaumkanten aus $G \setminus T$ in einer beliebigen Ordnung, wobei gilt $N = m - n + 1$.

Eingabe: Zusammenhängender Graph $G = (V, E)$

Ausgabe: MCB von G

for $j = 1$ *bis* N **do** $\qquad \mathcal{O}(m)$

  Initialisiere $S_{1,j} \leftarrow \{e_j\}$

for $k = 1$ *bis* N **do** $\qquad \mathcal{O}(m^3 + c \cdot m)$

  Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält

  for $j = k + 1$ *bis* N **do**

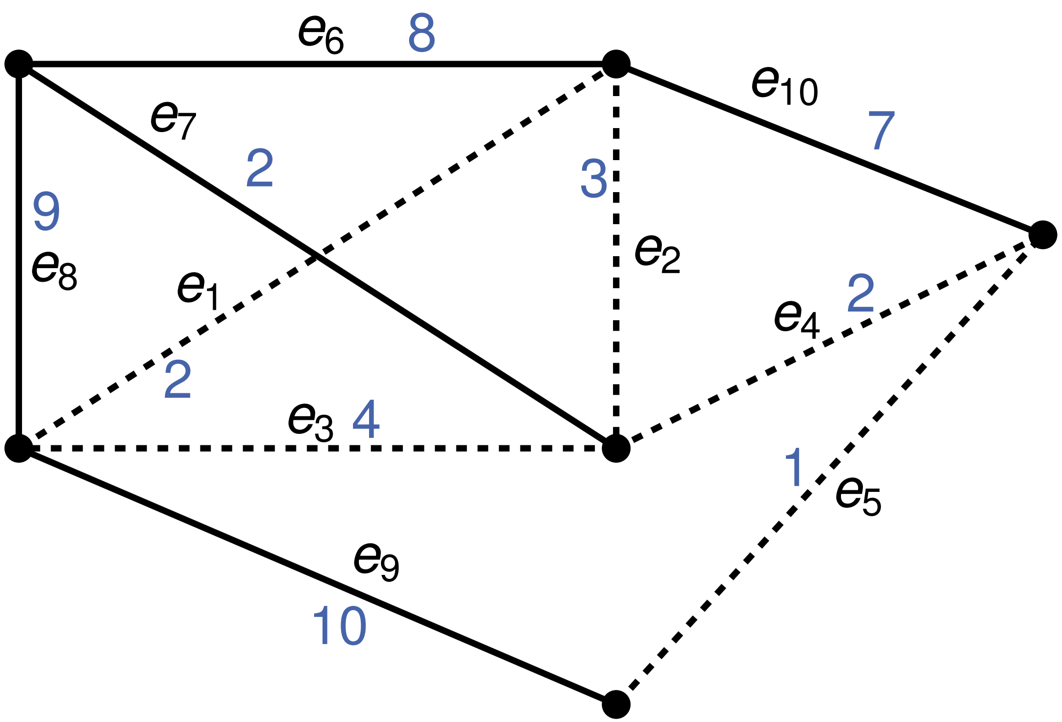
$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Ausgabe ist: $\{C_1, \ldots, C_N\}$

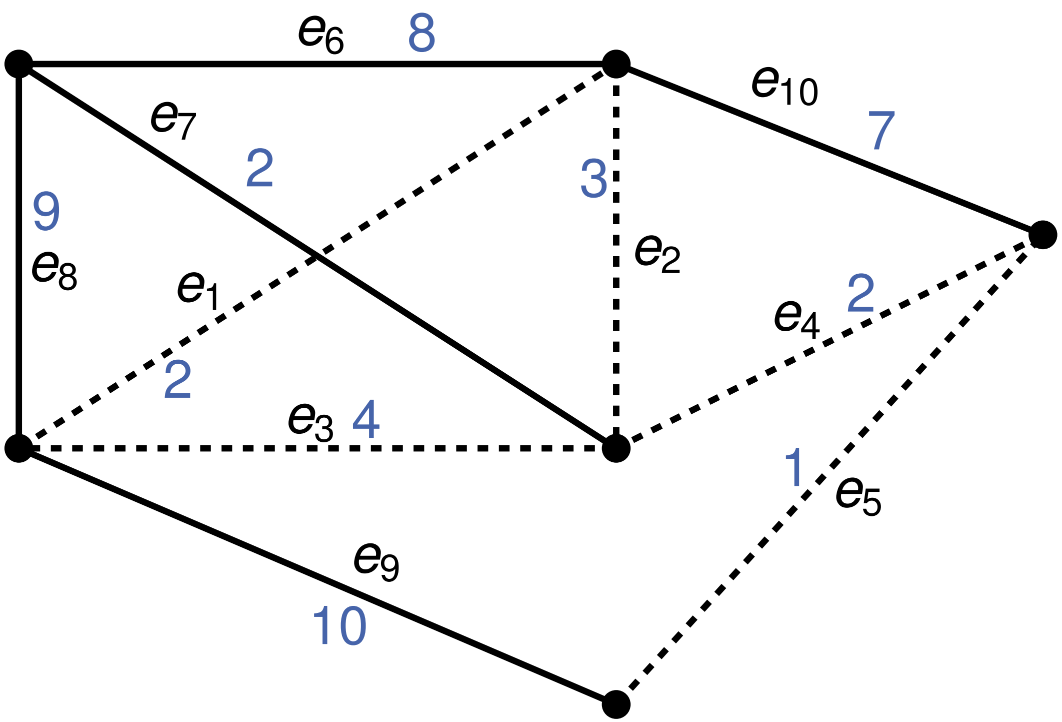
Annahme: Berechnung von C_k kann in $\mathcal{O}(c)$ durchgeführt werden.
Es gilt (ohne Beweis): $c \in \mathcal{O}(m^2 + n^2 \log n)$

Gesamtlaufzeit: $\mathcal{O}(m^3 + m \cdot n^2 \log n)$
vgl. Horton: $\mathcal{O}(m^3 \cdot n)$

Beispiel



Beispiel



Initialisierung:

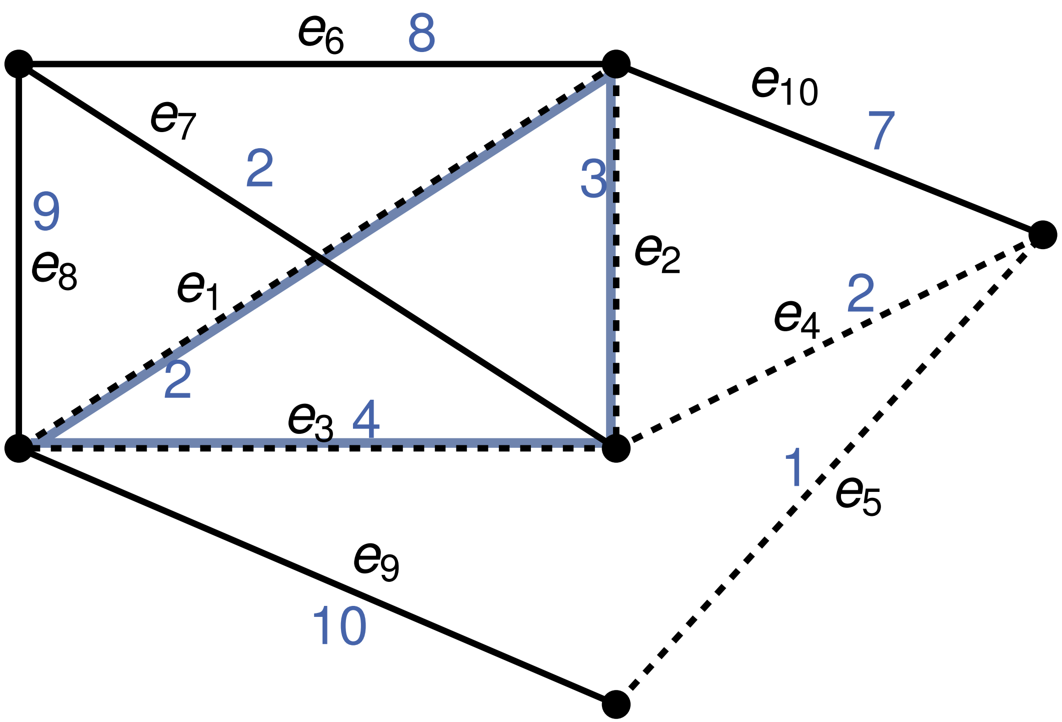
for j = 1 *bis* N **do**
Initialisiere $S_{1,j} \leftarrow \{e_j\}$

Ergebnis:

$S_{1,1} = \{e_1\}$, $S_{1,2} = \{e_2\}$, $S_{1,3} = \{e_3\}$, $S_{1,4} = \{e_4\}$, $S_{1,5} = \{e_5\}$

Beispiel

$S_{1,1} = \{e_1\}$, $S_{1,2} = \{e_2\}$, $S_{1,3} = \{e_3\}$,
$S_{1,4} = \{e_4\}$, $S_{1,5} = \{e_5\}$



k = 1: $S_{2,2} := S_{1,2} \oplus S_{1,1} = \{e_1, e_2\}$

$S_{2,3} := S_{1,3} \oplus S_{1,1} = \{e_1, e_3\}$

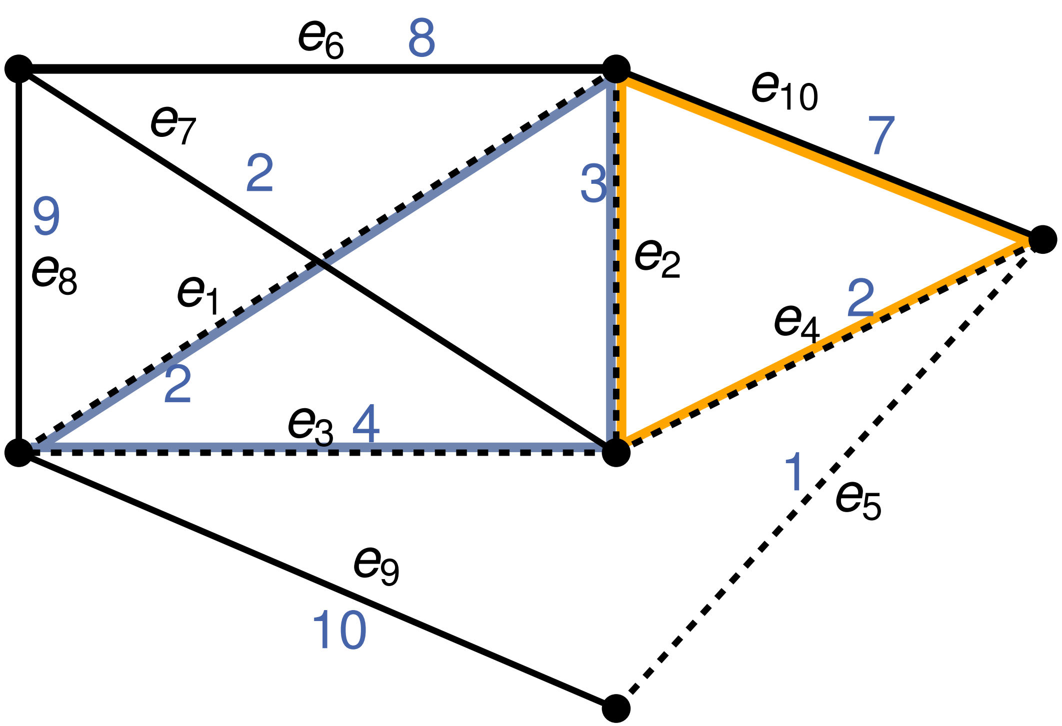
$S_{2,4} := \{e_4\}$

$S_{2,5} := \{e_5\}$ $\quad w(\mathbf{C}_1) = 9$

for $k = 1$ *bis* N **do**
- Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält
- **for** $j = k + 1$ *bis* N **do**
  - $$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Beispiel

$S_{1,1} = \{e_1\}$, $S_{1,2} = \{e_2\}$, $S_{1,3} = \{e_3\}$,
$S_{1,4} = \{e_4\}$, $S_{1,5} = \{e_5\}$



k = 1: $S_{2,2} := S_{1,2} \oplus S_{1,1} = \{e_1, e_2\}$
$S_{2,3} := S_{1,3} \oplus S_{1,1} = \{e_1, e_3\}$
$S_{2,4} := \{e_4\}$
$S_{2,5} := \{e_5\}$ $\quad w(\mathbf{C}_1) = 9$

k = 2: $S_{3,3} := S_{2,3} = \{e_1, e_3\}$
$S_{3,4} := S_{2,2} \oplus S_{2,4} = \{e_1, e_2, e_4\}$
$S_{3,5} := S_{2,5} = \{e_5\}$ $\quad w(\mathbf{C}_2) = 12$

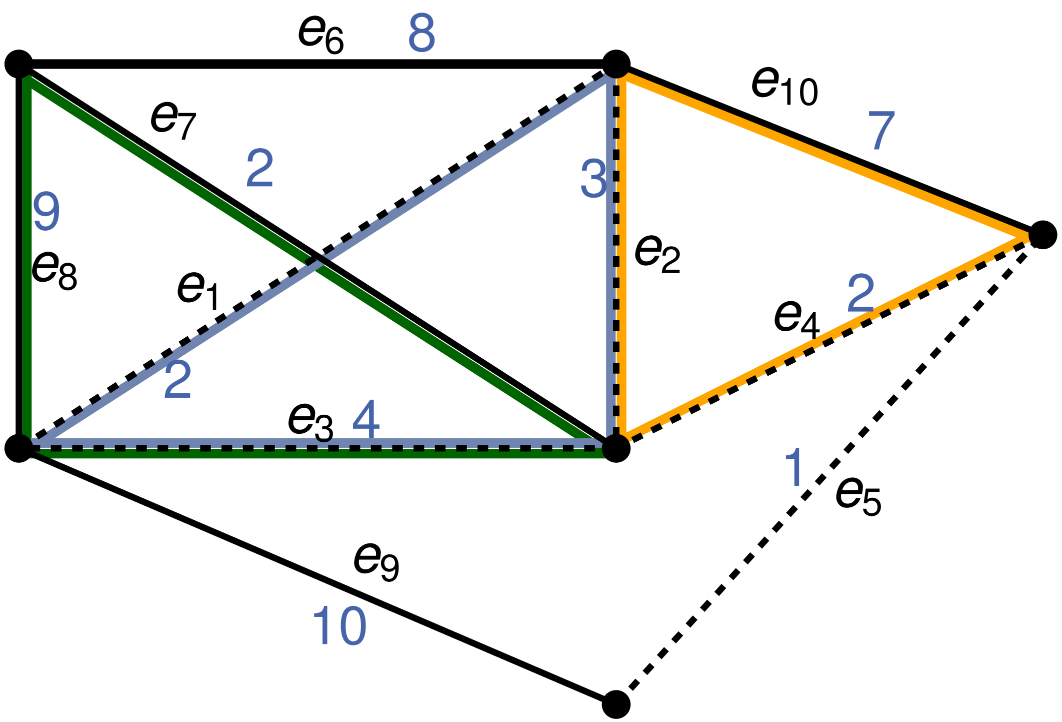
for $k = 1$ *bis* N **do**
  Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält
  for $j = k + 1$ *bis* N **do**

$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel

$S_{1,1} = \{e_1\}$, $S_{1,2} = \{e_2\}$, $S_{1,3} = \{e_3\}$,
$S_{1,4} = \{e_4\}$, $S_{1,5} = \{e_5\}$



k = 1: $S_{2,2} := S_{1,2} \oplus S_{1,1} = \{e_1, e_2\}$
$S_{2,3} := S_{1,3} \oplus S_{1,1} = \{e_1, e_3\}$
$S_{2,4} := \{e_4\}$
$S_{2,5} := \{e_5\}$ $\quad w(\mathbf{C}_1) = 9$

k = 2: $S_{3,3} := S_{2,3} = \{e_1, e_3\}$
$S_{3,4} := S_{2,2} \oplus S_{2,4} = \{e_1, e_2, e_4\}$
$S_{3,5} := S_{2,5} = \{e_5\}$ $\quad w(\mathbf{C}_2) = 12$

k = 3: $S_{4,4} := \{e_1, e_2, e_4\}$
$S_{4,5} := \{e_5\}$ $\quad w(\mathbf{C}_3) = 15$

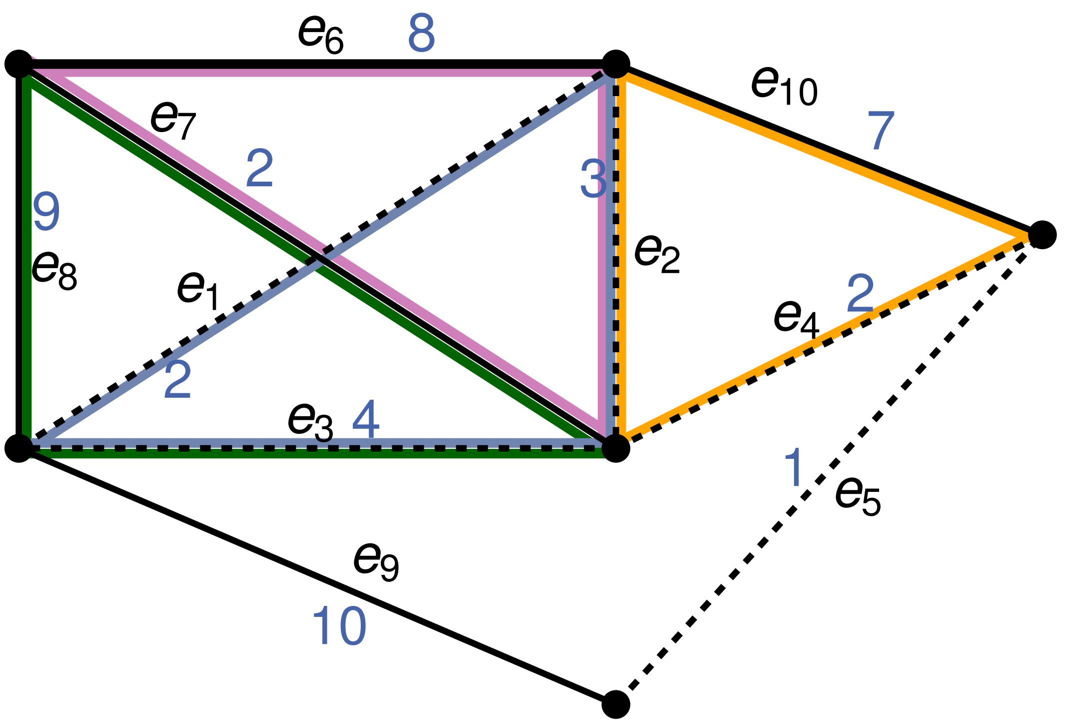
for $k = 1$ *bis* N **do**
  Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält
  for $j = k + 1$ *bis* N **do**

$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel

$S_{1,1} = \{e_1\}$, $S_{1,2} = \{e_2\}$, $S_{1,3} = \{e_3\}$,
$S_{1,4} = \{e_4\}$, $S_{1,5} = \{e_5\}$



k = 1: $S_{2,2} := S_{1,2} \oplus S_{1,1} = \{e_1, e_2\}$
$S_{2,3} := S_{1,3} \oplus S_{1,1} = \{e_1, e_3\}$
$S_{2,4} := \{e_4\}$
$S_{2,5} := \{e_5\}$ $\quad w(\mathbf{C}_1) = 9$

k = 2: $S_{3,3} := S_{2,3} = \{e_1, e_3\}$
$S_{3,4} := S_{2,2} \oplus S_{2,4} = \{e_1, e_2, e_4\}$
$S_{3,5} := S_{2,5} = \{e_5\}$ $\quad w(\mathbf{C}_2) = 12$

k = 3: $S_{4,4} := \{e_1, e_2, e_4\}$
$S_{4,5} := \{e_5\}$ $\quad w(\mathbf{C}_3) = 15$

k = 4: $S_{5,5} := \{e_5\}$ $\quad w(\mathbf{C}_4) = 13$

for $k = 1$ *bis* N **do**
  Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält
  for $j = k + 1$ *bis* N **do**

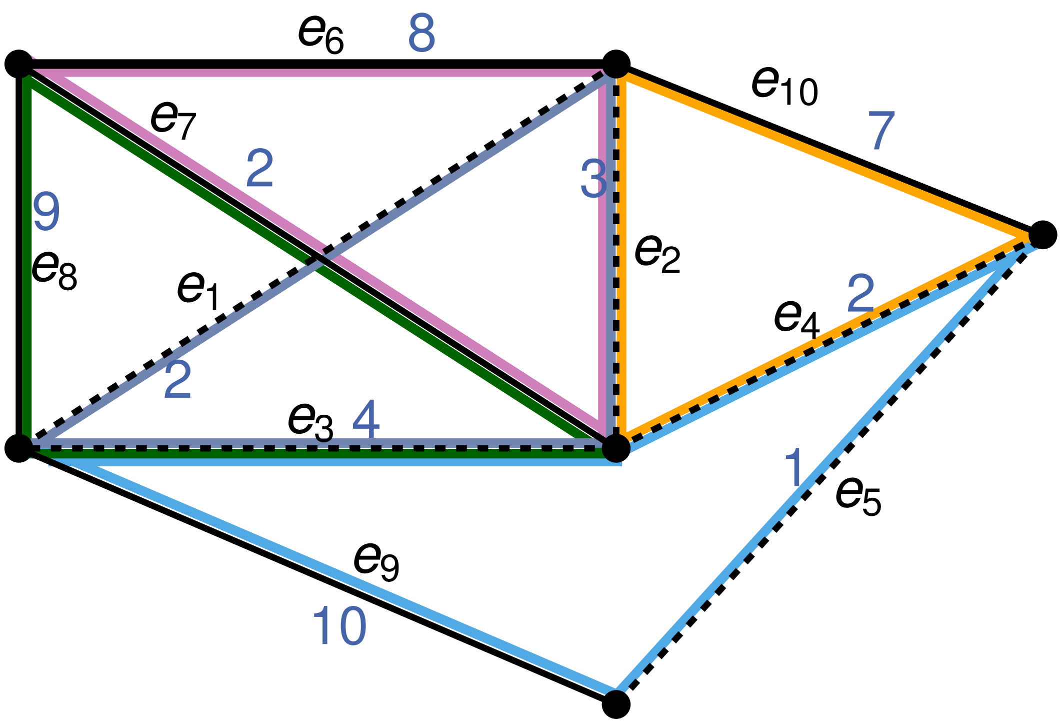
$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Beispiel

$S_{1,1} = \{e_1\},\ S_{1,2} = \{e_2\},\ S_{1,3} = \{e_3\},$
$S_{1,4} = \{e_4\},\ S_{1,5} = \{e_5\}$

Karlsruher Institut für Technologie



k = 1: $S_{2,2} := S_{1,2} \oplus S_{1,1} = \{e_1, e_2\}$
$S_{2,3} := S_{1,3} \oplus S_{1,1} = \{e_1, e_3\}$
$S_{2,4} := \{e_4\}$
$S_{2,5} := \{e_5\}$ $\quad w(\mathbf{C}_1) = 9$

k = 2: $S_{3,3} := S_{2,3} = \{e_1, e_3\}$
$S_{3,4} := S_{2,2} \oplus S_{2,4} = \{e_1, e_2, e_4\}$
$S_{3,5} := S_{2,5} = \{e_5\}$ $\quad w(\mathbf{C}_2) = 12$

k = 3: $S_{4,4} := \{e_1, e_2, e_4\}$
$S_{4,5} := \{e_5\}$ $\quad w(\mathbf{C}_3) = 15$

k = 4: $S_{5,5} := \{e_5\}$ $\quad w(\mathbf{C}_4) = 13$

k = 5: $\quad w(\mathbf{C}_5) = 17$

for $k = 1$ *bis* N **do**
  Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält
  for $j = k + 1$ *bis* N **do**

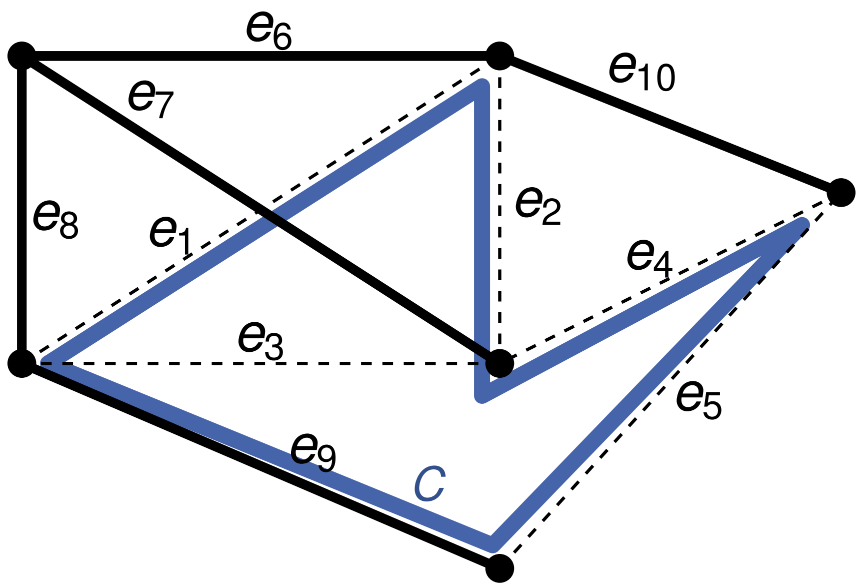
$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Korrektheit des Algorithmus von de Pina

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Vektorenschreibweise

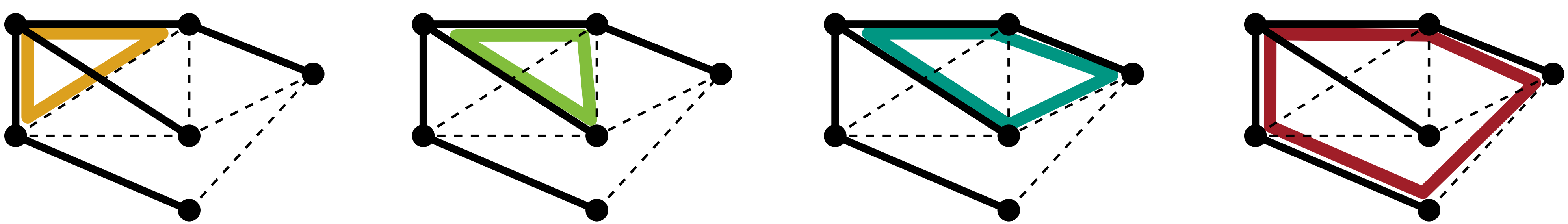
Beschreibe Kreise als Inzidenzvektoren über Nichtbaumkanten $\{e_1, \ldots, e_N\}$.



Beispiel: Kreise werden mithilfe der Nichtbaumkanten e_1, e_2, e_3, e_4 und e_5 beschrieben.

$$C = \begin{array}{cc} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} & \begin{array}{c} e_1 \\ \\ \vdots \\ \\ e_5 \end{array} \end{array}$$

Kreis C kann mithilfe der Fundamentalkreise C_i (C_i=Fundamentalkreis der Nichtbaumkante e_i) rekonstruiert werden. $C = C_1 \oplus C_2 \oplus C_4 \oplus C_5$



Bilinearform

for $k = 1$ *bis* N **do**

Finde einen kürzesten Kreis C_k, der eine ungerade Anzahl von Kanten aus $S_{k,k}$ enthält

for $j = k + 1$ *bis* N **do**

$$S_{k+1,j} \leftarrow \begin{cases} S_{k,j} & \text{, falls } C_k \text{ eine gerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \\ S_{k,j} \oplus S_{k,k} & \text{, falls } C_k \text{ eine ungerade Anzahl Kanten aus } S_{k,j} \text{ enthält} \end{cases}$$

Betrachte $S_{k,k}$ nach k-ten Durchlauf ebenfalls als Vektor über $\{e_1, \ldots, e_N\}$, der Menge der Nichtbaumkanten.

Definiere Bilinearform zweier Vektoren C und S: $\langle C, S \rangle := \sum_{i=1}^{N} (c_i \cdot s_i)$

Produkt und Summe sind über GF(2) definiert.

C und S sind *orthogonal* zueinander genau dann, wenn $\langle C, S \rangle = 0$.

$\langle C, S \rangle = 1$ genau dann, wenn C eine ungerade Anzahl Einträge mit S gemeinsam hat.

Algebraische Variante

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB von G

for $i = 1$ *bis* N **do**
    $S_i \leftarrow \{e_i\}$

for $k = 1$ *bis* N **do**
    Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$
    for $i = k + 1$ *bis* N **do**
        if $\langle C_k, S_i \rangle = 1$ **then**
            $S_i \leftarrow S_i \oplus S_k$

Ausgabe ist: $\{C_1, \ldots, C_N\}$

Hinweis: Schreibe S_k abkürzend für $S_{k,k}$

Invariante

```
for k = 1 bis N do
    Finde einen kürzesten Kreis C_k mit ⟨C_k, S_k⟩ = 1
    for i = k + 1 bis N do
        if ⟨C_k, S_i⟩ = 1 then
            S_i ← S_i ⊕ S_k
```

Lemma 5.12 Die zweite äußere Schleife des Algorithmus erhält die Invariante $\langle C_i, S_{j+1}\rangle = 0$ für alle i, $1 \leq i \leq j \leq N$.

Beweis: Zeige durch Induktion über Anzahl k an Durchläufen, dass

$$\langle C_i, S_j\rangle = 0 \text{ für alle } i \text{ mit } 1 \leq i \leq k \text{ und } j \text{ mit } k < j \leq N.$$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Invariante

Karlsruher Institut für Technologie

```
1 for k = 1 bis N do
2 |   Finde einen kürzesten Kreis C_k mit ⟨C_k, S_k⟩ = 1
3 |   for i = k + 1 bis N do
4 |   |   if ⟨C_k, S_i⟩ = 1 then
5 |   |   |   S_i ← S_i ⊕ S_k
```

Lemma 5.12 Die zweite äußere Schleife des Algorithmus erhält die Invariante $\langle C_i, S_{j+1} \rangle = 0$ für alle i, $1 \leq i \leq j \leq N$.

Beweis: Zeige durch Induktion über Anzahl k an Durchläufen, dass

$$\langle C_i, S_j \rangle = 0 \text{ für alle } i \text{ mit } 1 \leq i \leq k \text{ und } j \text{ mit } k < j \leq N.$$

IA: $k = 1$ Betrachte S_j, das nicht orthogonal zu C_i ist.

S_j wird in Zeile 5 orthogonalisiert, denn nach Addition gilt:

$$\langle C_1, S_j^{neu} \rangle = \langle C_1, S_j \oplus S_1 \rangle = \langle C_1, S_j \rangle + \langle C_1, S_1 \rangle = 1 + 1 = 0$$

Invariante

```
1 for k = 1 bis N do
2     Finde einen kürzesten Kreis C_k mit ⟨C_k, S_k⟩ = 1
3     for i = k + 1 bis N do
4         if ⟨C_k, S_i⟩ = 1 then
5             S_i ← S_i ⊕ S_k
```

Lemma 5.12 Die zweite äußere Schleife des Algorithmus erhält die Invariante $\langle C_i, S_{j+1} \rangle = 0$ für alle i, $1 \leq i \leq j \leq N$.

Beweis: Zeige durch Induktion über Anzahl k an Durchläufen, dass

$$\langle C_i, S_j \rangle = 0 \text{ für alle } i \text{ mit } 1 \leq i \leq k \text{ und } j \text{ mit } k < j \leq N.$$

IS: $2 \leq k \leq N$ Betrachte die Kreise $C_1, \ldots, C_k$ und sei $S_{i,j}$ der Zeuge S_j nach dem i-ten Durchlauf.

1. Fall: Es gilt $S_{k,j} = S_{k-1,j}$

- Wegen Tests in Zeile 4 gilt: $\langle C_k, S_{k-1,j} \rangle = 0 = \langle C_k, S_{k,j} \rangle$
- Wegen Induktionsvoraussetzung gilt: $\langle C_i, S_{k-1,j} \rangle = 0$ für $i < k$

2. Fall: Es gilt $S_{k,j} = S_{k-1,j} \oplus S_{k,k}$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Invariante

for $k = 1$ *bis* N **do**
  Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$
  for $i = k + 1$ *bis* N **do**
    if $\langle C_k, S_i \rangle = 1$ **then**
      $S_i \leftarrow S_i \oplus S_k$

Lemma 5.12 Die zweite äußere Schleife des Algorithmus erhält die Invariante $\langle C_i, S_{j+1} \rangle = 0$ für alle i, $1 \leq i \leq j \leq N$.

Beweis: Zeige durch Induktion über Anzahl k an Durchläufen, dass

$$\langle C_i, S_j \rangle = 0 \text{ für alle } i \text{ mit } 1 \leq i \leq k \text{ und } j \text{ mit } k < j \leq N.$$

IS: $2 \leq k \leq N$ Betrachte die Kreise $C_1, \ldots, C_k$ und sei $S_{i,j}$ der Zeuge S_j nach dem i-ten Durchlauf.

1. Fall: Es gilt $S_{k,j} = S_{k-1,j}$

2. Fall: Es gilt $S_{k,j} = S_{k-1,j} \oplus S_{k,k}$

Es gilt $\langle C_k, S_{k-1,j} \rangle = 1$

Invariante

for $k = 1$ *bis* N **do**
    Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$
    for $i = k + 1$ *bis* N **do**
        if $\langle C_k, S_i \rangle = 1$ **then**
            $S_i \leftarrow S_i \oplus S_k$

Lemma 5.12 Die zweite äußere Schleife des Algorithmus erhält die Invariante $\langle C_i, S_{j+1} \rangle = 0$ für alle i, $1 \leq i \leq j \leq N$.

Beweis: Zeige durch Induktion über Anzahl k an Durchläufen, dass

$$\langle C_i, S_j \rangle = 0 \text{ für alle } i \text{ mit } 1 \leq i \leq k \text{ und } j \text{ mit } k < j \leq N.$$

IS: $2 \leq k \leq N$ Betrachte die Kreise $C_1, \ldots, C_k$ und sei $S_{i,j}$ der Zeuge S_j nach dem i-ten Durchlauf.

1. Fall: Es gilt $S_{k,j} = S_{k-1,j}$

2. Fall: Es gilt $S_{k,j} = S_{k-1,j} \oplus S_{k,k}$

Es gilt $\langle C_k, S_{k-1,j} \rangle = 1$

Betrachte für $1 \leq \mathbf{i} < \mathbf{k} < \mathbf{j} \leq \mathbf{N}$**:** $\langle C_i, S_{k,j} \rangle = \langle C_i, S_{k-1,j} \oplus S_{k,k} \rangle = \langle C_i, S_{k-1,j} \rangle + \langle C_i, S_{k,k} \rangle$

Wegen Induktionsvoraussetzung gilt: $\langle C_i, S_{k-1,j} \rangle = 0$

Da stets $S_{k,k} = S_{k-1,k}$, gilt nach Induktionsvoraussetzung: $\langle C_i, S_{k,k} \rangle = \langle C_i, S_{k-1,k} \rangle = 0$

Invariante


```
for k = 1 bis N do
│   Finde einen kürzesten Kreis C_k mit ⟨C_k, S_k⟩ = 1
│   for i = k + 1 bis N do
│   │   if ⟨C_k, S_i⟩ = 1 then
│   │   │   S_i ← S_i ⊕ S_k
```

Lemma 5.12 Die zweite äußere Schleife des Algorithmus erhält die Invariante $\langle C_i, S_{j+1} \rangle = 0$ für alle i, $1 \leq i \leq j \leq N$.

Beweis: Zeige durch Induktion über Anzahl k an Durchläufen, dass

$$\langle C_i, S_j \rangle = 0 \text{ für alle } i \text{ mit } 1 \leq i \leq k \text{ und } j \text{ mit } k < j \leq N.$$

IS: $2 \leq k \leq N$ Betrachte die Kreise $C_1, \ldots, C_k$ und sei $S_{i,j}$ der Zeuge S_j nach dem i-ten Durchlauf.

1. Fall: Es gilt $S_{k,j} = S_{k-1,j}$

2. Fall: Es gilt $S_{k,j} = S_{k-1,j} \oplus S_{k,k}$

Es gilt $\langle C_k, S_{k-1,j} \rangle = 1$

Betrachte für $1 \leq \mathbf{i} < \mathbf{k} < \mathbf{j} \leq \mathbf{N}$: $\langle C_i, S_{k,j} \rangle = \langle C_i, S_{k-1,j} \oplus S_{k,k} \rangle = \langle C_i, S_{k-1,j} \rangle + \langle C_i, S_{k,k} \rangle$

Wegen Induktionsvoraussetzung gilt: $\langle C_i, S_{k-1,j} \rangle = 0$

Da stets $S_{k,k} = S_{k-1,k}$, gilt nach Induktionsvoraussetzung: $\langle C_i, S_{k,k} \rangle = \langle C_i, S_{k-1,k} \rangle = 0$

Für i=k gilt: $\langle C_k, S_{k,j} \rangle = \langle C_k, S_{k-1,j} \rangle + \langle C_k, S_{k,k} \rangle = 1 + 1 = 0$

Vereinfachende Schreibweise: SIMPLE MCB

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB= $\{C_1, \dots, C_N\}$ von G

$S_1 \leftarrow \{e_1\}$; $C_1 \leftarrow$ kürzester Kreis mit $\langle C_1, S_1 \rangle = 1$

for $k = 2$ *bis* N **do**

  Berechne Vektor S_k, der nichttriviale Lösung des Systems $\langle C_i, X \rangle = 0$ für $i = 1$ bis $k - 1$ ist.

  Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$.

Satz 5.13 Der Algorithmus SIMPLE MCB berechnet eine MCB.

1. $\{C_1, \dots, C_N\}$ ist Basis.
2. $\{C_1, \dots, C_N\}$ hat minimales Gewicht.

Vereinfachende Schreibweise: SIMPLE MCB

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB= $\{C_1, \ldots, C_N\}$ von G

$S_1 \leftarrow \{e_1\}$; $C_1 \leftarrow$ kürzester Kreis mit $\langle C_1, S_1 \rangle = 1$

for $k = 2$ *bis* N **do**

  Berechne Vektor S_k, der nichttriviale Lösung des Systems $\langle C_i, X \rangle = 0$ für $i = 1$ bis $k - 1$ ist.

  Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$.

Satz 5.13 Der Algorithmus SIMPLE MCB berechnet eine MCB.

1. $\{C_1, \ldots, C_N\}$ ist Basis.

Da $\langle C_i, S_k \rangle = 0$ für $1 \leq i \leq k - 1$ und $\langle C_k, S_k \rangle = 1$, ist C_k lin. unab. von $\{C_1, \ldots, C_{k-1}\}$

$\Rightarrow$ $\{C_1, \ldots, C_N\}$ ist eine Basis.

2. $\{C_1, \ldots, C_N\}$ hat minimales Gewicht.

Vereinfachende Schreibweise: SIMPLE MCB

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB= $\{C_1, \dots, C_N\}$ von G

$S_1 \leftarrow \{e_1\}$; $C_1 \leftarrow$ kürzester Kreis mit $\langle C_1, S_1 \rangle = 1$

for $k = 2$ *bis* N **do**

   Berechne Vektor S_k, der nichttriviale Lösung des Systems $\langle C_i, X \rangle = 0$ für $i = 1$ bis $k - 1$ ist.

   Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$.

Satz 5.13 Der Algorithmus SIMPLE MCB berechnet eine MCB.

1. $\{C_1, \dots, C_N\}$ ist Basis.
2. $\{C_1, \dots, C_N\}$ hat minimales Gewicht.

Annahme: $\{C_1, \dots, C_N\}$ ist nicht MCB. Sei $\mathcal{B}$ eine MCB.

Wähle i so, dass $\{C_1, \dots, C_i\} \subseteq \mathcal{B}$, aber für keine MCB $\mathcal{B}'$ gilt $\{C_1, \dots, C_{i+1}\} \subseteq \mathcal{B}'$

Vereinfachende Schreibweise: SIMPLE MCB

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB= $\{C_1, \ldots, C_N\}$ von G

$S_1 \leftarrow \{e_1\}$; $C_1 \leftarrow$ kürzester Kreis mit $\langle C_1, S_1\rangle = 1$

for $k = 2$ *bis* N **do**

- Berechne Vektor S_k, der nichttriviale Lösung des Systems $\langle C_i, X\rangle = 0$ für $i = 1$ bis $k - 1$ ist.
- Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k\rangle = 1$.

Satz 5.13 Der Algorithmus SIMPLE MCB berechnet eine MCB.

1. $\{C_1, \ldots, C_N\}$ ist Basis.
2. $\{C_1, \ldots, C_N\}$ hat minimales Gewicht.

Annahme: $\{C_1, \ldots, C_N\}$ ist nicht MCB. Sei $\mathcal{B}$ eine MCB.

Wähle i so, dass $\{C_1, \ldots, C_i\} \subseteq \mathcal{B}$, aber für keine MCB $\mathcal{B}'$ gilt $\{C_1, \ldots, C_{i+1}\} \subseteq \mathcal{B}'$

Da $\mathcal{B}$ Basis, existieren $D_1, \ldots, D_\ell \in \mathcal{B}$ mit $C_{i+1} = D_1 \oplus \cdots \oplus D_\ell$

Nach Konstruktion: $\langle C_{i+1}, S_{i+1}\rangle = 1 \rightarrow$ es existiert D_j mit $\langle D_j, S_{i+1}\rangle = 1$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Vereinfachende Schreibweise: SIMPLE MCB

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB= $\{C_1, \ldots, C_N\}$ von G

$S_1 \leftarrow \{e_1\}$; $C_1 \leftarrow$ kürzester Kreis mit $\langle C_1, S_1 \rangle = 1$

for $k = 2$ *bis* N **do**

Berechne Vektor S_k, der nichttriviale Lösung des Systems $\langle C_i, X \rangle = 0$ für $i = 1$ bis $k - 1$ ist.

Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$.

Satz 5.13 Der Algorithmus SIMPLE MCB berechnet eine MCB.

1. $\{C_1, \ldots, C_N\}$ ist Basis.
2. $\{C_1, \ldots, C_N\}$ hat minimales Gewicht.

Annahme: $\{C_1, \ldots, C_N\}$ ist nicht MCB. Sei $\mathcal{B}$ eine MCB.

Wähle i so, dass $\{C_1, \ldots, C_i\} \subseteq \mathcal{B}$, aber für keine MCB $\mathcal{B}'$ gilt $\{C_1, \ldots, C_{i+1}\} \subseteq \mathcal{B}'$

Da $\mathcal{B}$ Basis, existieren $D_1, \ldots, D_\ell \in \mathcal{B}$ mit $C_{i+1} = D_1 \oplus \cdots \oplus D_\ell$

Nach Konstruktion: $\langle C_{i+1}, S_{i+1} \rangle = 1 \rightarrow$ es existiert D_j mit $\langle D_j, S_{i+1} \rangle = 1$

Da C_{i+1} kürzester Kreis mit $\langle C_{i+1}, S_{i+1} \rangle$ ist, gilt $w(C_{i+1}) \leq w(D_j)$.

Setze $\mathcal{B}^\star := \mathcal{B} \setminus \{D_j\} \cup \{C_{i+1}\}$, $\mathcal{B}^\star$ ist wieder MCB.

Vereinfachende Schreibweise: SIMPLE MCB

Eingabe: Graph $G = (V, E)$

Ausgabe: MCB= $\{C_1, \ldots, C_N\}$ von G

$S_1 \leftarrow \{e_1\}$; $C_1 \leftarrow$ kürzester Kreis mit $\langle C_1, S_1 \rangle = 1$

for $k = 2$ *bis* N **do**

  Berechne Vektor S_k, der nichttriviale Lösung des Systems $\langle C_i, X \rangle = 0$ für $i = 1$ bis $k - 1$ ist.

  Finde einen kürzesten Kreis C_k mit $\langle C_k, S_k \rangle = 1$.

Satz 5.13 Der Algorithmus SIMPLE MCB berechnet eine MCB.

1. $\{C_1, \ldots, C_N\}$ ist Basis.
2. $\{C_1, \ldots, C_N\}$ hat minimales Gewicht.

Annahme: $\{C_1, \ldots, C_N\}$ ist nicht MCB. Sei $\mathcal{B}$ eine MCB.

Wähle i so, dass $\{C_1, \ldots, C_i\} \subseteq \mathcal{B}$, aber für keine MCB $\mathcal{B}'$ gilt $\{C_1, \ldots, C_{i+1}\} \subseteq \mathcal{B}'$

Da $\mathcal{B}$ Basis, existieren $D_1, \ldots, D_\ell \in \mathcal{B}$ mit $C_{i+1} = D_1 \oplus \cdots \oplus D_\ell$

Nach Konstruktion: $\langle C_{i+1}, S_{i+1} \rangle = 1 \rightarrow$ es existiert D_j mit $\langle D_j, S_{i+1} \rangle = 1$

Da C_{i+1} kürzester Kreis mit $\langle C_{i+1}, S_{i+1} \rangle$ ist, gilt $w(C_{i+1}) \leq w(D_j)$.

Setze $\mathcal{B}^\star := \mathcal{B} \setminus \{D_j\} \cup \{C_{i+1}\}$, $\mathcal{B}^\star$ ist wieder MCB.

Da $\langle D_j, S_{i+1} \rangle = 1$ und $\langle C_j, S_{i+1} \rangle = 0$ für $1 \leq j \leq i$ gilt $D_j \notin \{C_1, \ldots, C_i\}$.

Damit: $\mathcal{B}^\star$ ist MCB mit $\{C_1, \ldots, C_{i+1}\} \subseteq \mathcal{B}^\star \rightarrow$ Widerspruch zur Wahl von i.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

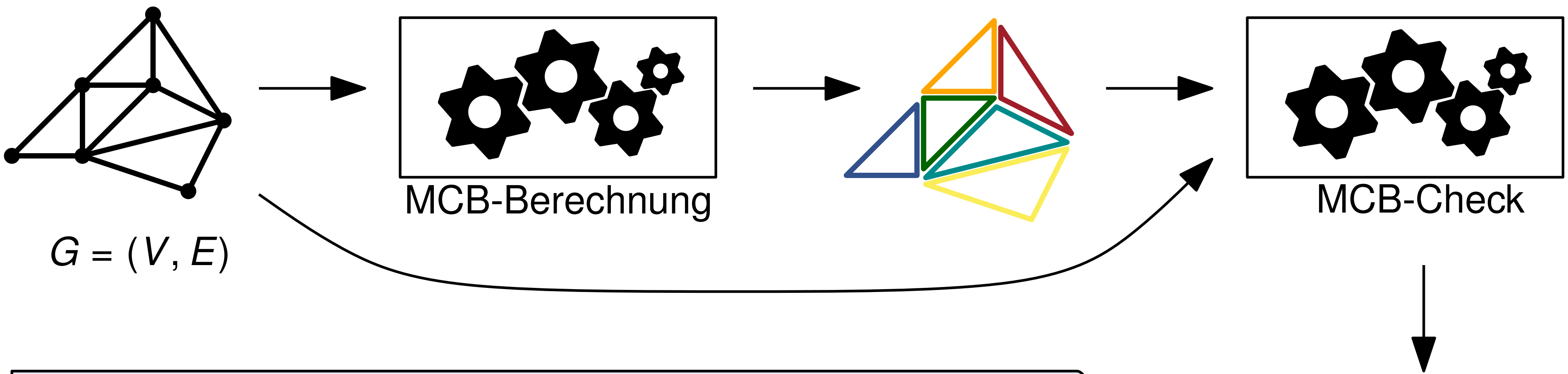
Bemerkungen

- Die Laufzeit kann auf $\mathcal{O}(m^2 \cdot n + m \cdot n^2 \cdot \log n)$ reduziert werden.
- Empirische Verbesserung: *Verheiratung* von Horton und de Pina.
  - Berechne Kandiatenmenge $\mathcal{H}$ von Horton.
  - Suche kürzesten Kreis C_k ausschließlich in dieser Kandiatenmenge $\rightarrow$ Lösungsraum wird verkleinert.
- Algorithmus von Horton kann mithilfe schneller Matrix-Multiplikation auf eine Laufzeit $\mathcal{O}(m^\omega n)$ reduziert werden (Bekannt: $\omega < 2.376$).

Zertifikat für MCB

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Zertifikat



Problem: Zertifikat für MCB-Algorithmus
Gegeben sei der Graph $G = (V, E)$, $w\colon E \to \mathbb{R}_0^+$ und eine Menge von Kreisen $\mathcal{A}$ von G. Gib ein Zertifikat dafür an, dass $\mathcal{A}$ eine MCB von G ist.

Zertifikat

Die gegebenen Kreise bilden eine

minimale Kreisbasis

in $G = (V, E)$.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Zertifikat-Erstellung

MCB-CHECKER

Eingabe: Graph $G = (V, E)$, Kreise $C_1, \dots C_N$

Ausgabe: Zertifikat zur Prüfung, ob $C_1, \dots C_N$ eine MCB von G sind.

1. Berechne aufspannenden Wald T, dabei seien $\{e_1, \dots, e_N\}$ die Nichtbaumkanten.
2. Definiere $A := (C_1 \dots C_N)$ als $N \times N$-Matrix, deren i-te Spalte der Inzidenzvektor von C_i mit $\{e_1, \dots, e_N\}$ ist.
3. Berechne A^{-1}.

Lemma 5.16 Seien $S_1, \dots, S_N$ und $C_1, \dots, C_N$ linear unabhängig. Falls C_i ein kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$ ist, dann ist $C_1, \dots, C_N$ eine MCB. **Ohne Beweis.**

Zertifikat-Erstellung

MCB-CHECKER

Eingabe: Graph $G = (V, E)$, Kreise $C_1, \dots C_N$

Ausgabe: Zertifikat zur Prüfung, ob $C_1, \dots C_N$ eine MCB von G sind.

1. Berechne aufspannenden Wald T, dabei seien $\{e_1, \dots, e_N\}$ die Nichtbaumkanten.
2. Definiere $A := (C_1 \dots C_N)$ als $N \times N$-Matrix, deren i-te Spalte der Inzidenzvektor von C_i mit $\{e_1, \dots, e_N\}$ ist.
3. Berechne A^{-1}.

Lemma 5.16 Seien $S_1, \dots, S_N$ und $C_1, \dots, C_N$ linear unabhängig. Falls C_i ein kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$ ist, dann ist $C_1, \dots, C_N$ eine MCB. **Ohne Beweis.**

Folgerung: Angenommen A ist invertierbar.

$\longrightarrow$ $C_1, \dots, C_N$ linear unabhängig.

$\longrightarrow$ Zeilen $S_1, \dots, S_N$ von A^{-1} auch linear unabhängig.

$\longrightarrow$ $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$

Zertifikat-Erstellung

MCB-CHECKER

Eingabe: Graph $G = (V, E)$, Kreise $C_1, \dots C_N$

Ausgabe: Zertifikat zur Prüfung, ob $C_1, \dots C_N$ eine MCB von G sind.

1. Berechne aufspannenden Wald T, dabei seien $\{e_1, \dots, e_N\}$ die Nichtbaumkanten.
2. Definiere $A := (C_1 \dots C_N)$ als $N \times N$-Matrix, deren i-te Spalte der Inzidenzvektor von C_i mit $\{e_1, \dots, e_N\}$ ist.
3. Berechne A^{-1}.

Lemma 5.16 Seien $S_1, \dots, S_N$ und $C_1, \dots, C_N$ linear unabhängig. Falls C_i ein kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$ ist, dann ist $C_1, \dots, C_N$ eine MCB. **Ohne Beweis.**

Folgerung: Angenommen A ist invertierbar.

$\longrightarrow$ $C_1, \dots, C_N$ linear unabhängig.

$\longrightarrow$ Zeilen $S_1, \dots, S_N$ von A^{-1} auch linear unabhängig.

$\longrightarrow$ $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$

Wenn C_i also kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, dann ist $\{C_1, \dots, C_N\}$ eine MCB.

Falls Kreis C_i nicht kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, dann lässt sich der Kreis C_i durch einen kürzeren Kreis C_i mit $\langle S_i, C_i \rangle = 1$ ersetzen (siehe Korrektheitsbeweis SIMPLE MCB).

Zertifikat-Erstellung

MCB-CHECKER

Eingabe: Graph $G = (V, E)$, Kreise $C_1, \dots C_N$

Ausgabe: Zertifikat zur Prüfung, ob $C_1, \dots C_N$ eine MCB von G sind.

1. Berechne aufspannenden Wald T, dabei seien $\{e_1, \dots, e_N\}$ die Nichtbaumkanten.
2. Definiere $A := (C_1 \dots C_N)$ als $N \times N$-Matrix, deren i-te Spalte der Inzidenzvektor von C_i mit $\{e_1, \dots, e_N\}$ ist.
3. Berechne A^{-1}.

Lemma 5.16 Seien $S_1, \dots, S_N$ und $C_1, \dots, C_N$ linear unabhängig. Falls C_i ein kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$ ist, dann ist $C_1, \dots, C_N$ eine MCB. **Ohne Beweis.**

Folgerung: Angenommen A ist invertierbar.

$\longrightarrow$ $C_1, \dots, C_N$ linear unabhängig.

$\longrightarrow$ Zeilen $S_1, \dots, S_N$ von A^{-1} auch linear unabhängig.

$\longrightarrow$ $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$

Wenn C_i also kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, dann ist $\{C_1, \dots, C_N\}$ eine MCB.

Falls Kreis C_i nicht kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, dann lässt sich der Kreis C_i durch einen kürzeren Kreis C_i mit $\langle S_i, C_i \rangle = 1$ ersetzen (siehe Korrektheitsbeweis SIMPLE MCB).

$\{C_1, \dots, C_N\}$ ist genau dann MCB, wenn C_i kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, für alle $1 \leq i \leq N$.

Zertifikat-Erstellung

MCB-CHECKER

Eingabe: Graph $G = (V, E)$, Kreise $C_1, \dots C_N$

Ausgabe: Zertifikat zur Prüfung, ob $C_1, \dots C_N$ eine MCB von G sind.

1. Berechn … chtbaumkanten.

2. Definiere … nzidenzvektor von C_i mit $\{e_1,$ …

3. Berechn …

Lemma 5.16 … lls C_i ein kürzester Kreis mit $\langle S_i,$ … B. **Ohne Beweis.**

Zertifikat

Die Zeilen $S_1, \dots, S_N$ von A^{-1} bilden Zertifikat, dass $\{C_1, \dots, C_N\}$ eine MCB ist.

Folgerung: An …

→ …

→ Zeilen $S_1, \dots, S_N$ von A^{-1} auch linear unabhängig.

→ $\langle S_i, C_i \rangle = 1$ für alle $1 \leq i \leq N$

Wenn C_i also kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, dann ist $\{C_1, \dots, C_N\}$ eine MCB.

Falls Kreis C_i nicht kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, dann lässt sich der Kreis C_i durch einen kürzeren Kreis C_i mit $\langle S_i, C_i \rangle = 1$ ersetzen (siehe Korrektheitsbeweis SIMPLE MCB).

$\{C_1, \dots, C_N\}$ ist genau dann MCB, wenn C_i kürzester Kreis mit $\langle S_i, C_i \rangle = 1$ ist, für alle $1 \leq i \leq N$.