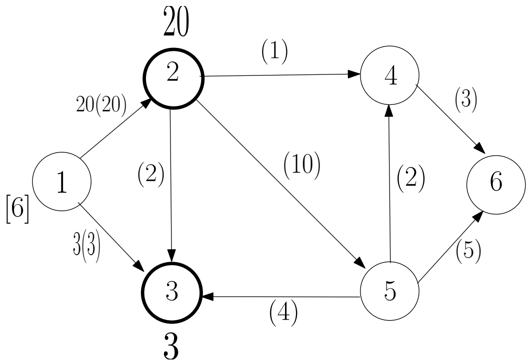
Universität Karlsruhe
Fakultät für Informatik
ITI Wagner

4. Musterlösung

Problem 1: Goldberg–Tarjan 1pt

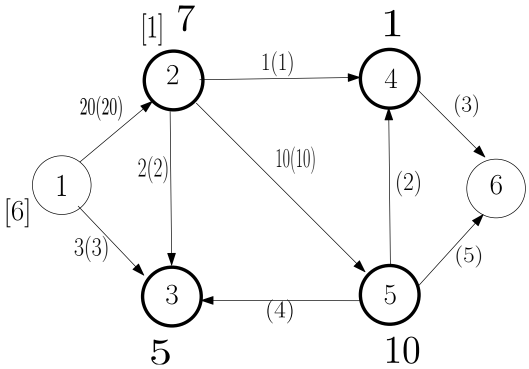
Die erste Darstellung zeigt das Netzwerk zu Beginn des Algorithmus



Wir führen jetzt die folgenden Operationen durch:

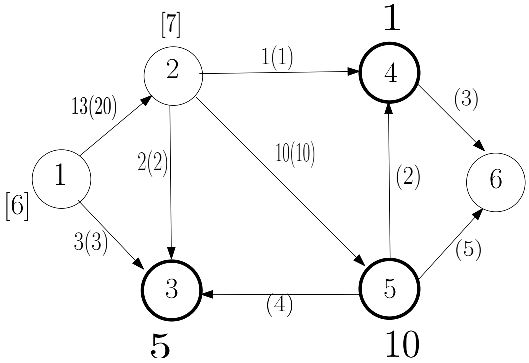
- Relabel(2). Dann ist $dist(2) = 1$.
- Push(2,3) mit $\Delta = 2$ und Push(2,4) mit $\Delta = 1$ und Push(2,5) mit $\Delta = 10$. Dann ist $e(2) = 7$, $e(3) = 5$, $e(4) = 1$,$e(5) = 10$

  Der neue Zwischenstand:



- Relabel(2). Da $r_f(2,1) > 0$ und $dist(2) < dist(1)$ ist, folgt $dist(2) := dist(1) + 1 := 7$
- Push(2,1) mit $\Delta = 7$ Dann $f(1,2) = 13$ und $e(2) = 0$

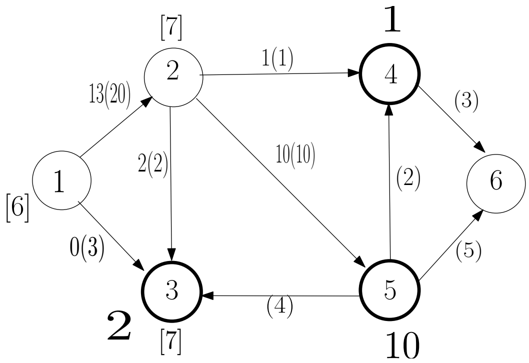
Jetzt haben wir den folgenden Zwischenstand im Netzwerk:



Nun folgen diese Operationen

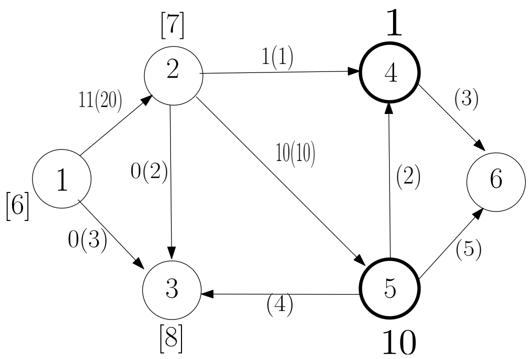
- RELABEL(3). Da $r_f(3,1) > 0$ und $dist(3) < dist(1)$ ist, folgt $dist(3) := dist(1) + 1 := 7$
- PUSH(3,1) mit $\Delta = 3$. Dann ist $e(3) = 2$

  Der neue Zwischenstand:



- RELABEL(3). Da $r_f(3,2) > 0$ und $dist(3) < dist(2)$ ist, folgt $dist(3) := dist(2) + 1 := 8$
- PUSH(3,2) mit $\Delta = 2$. Dann $e(2) = 2$ und $e(3) = 0$
- PUSH(2,1) mit $\Delta = 2$. Dann $e(2) = 0$

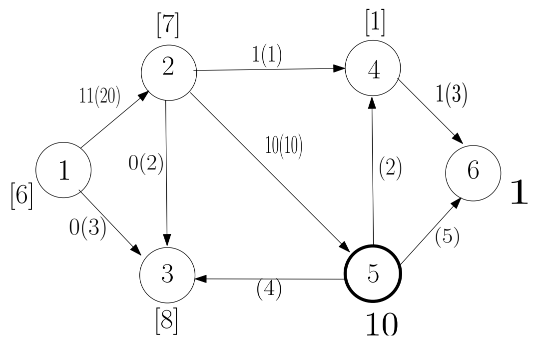
Der neue Zwischenstand sieht so aus:



Weiter:

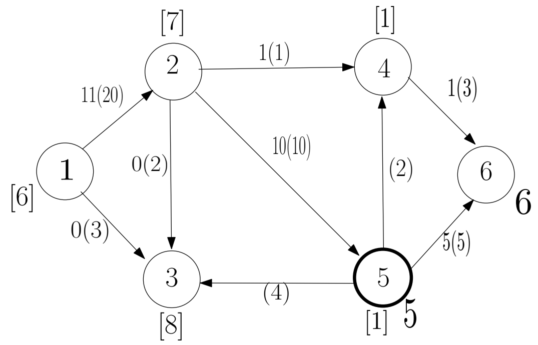
- RELABEL(4), folgt $dist(4) := 1$. PUSH(4,6) mit $\Delta = 1$.Dann ist $e(4) = 0$ und $e(6) = 1$

Der neue Zwischenstand:



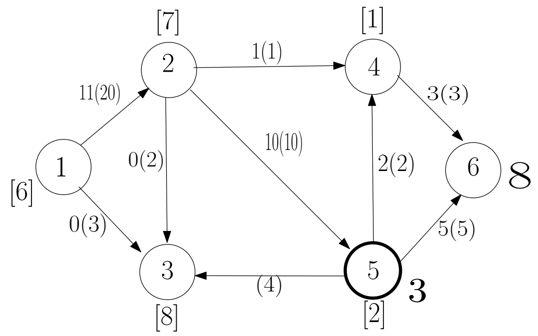
- Relabel(5) folgt $dist(5) := 1$. Push(5,6) mit $\Delta = 5$.Dann $e(5) = 5$ und $e(6) = 6$

  Der neue Zwischenstand:



- Relabel(5),folgt $dist(5) := dist(4) + 1 = 2$. Push(5,4) mit $\Delta = 2$.Dann ist $e(5) = 3$ und $e(4) = 2$
- Push(4,6) mit $\Delta = 2$. Dann ist $e(4) = 0$ und $e(6) = 8$

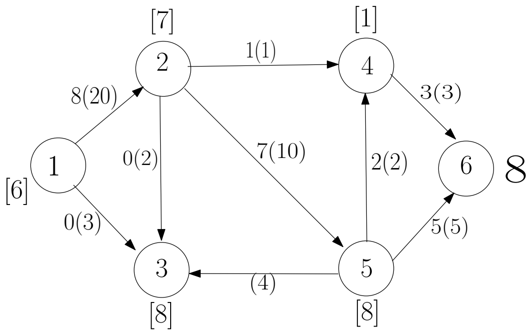
Das Netwerk sieht dann so aus:



Zum Schluss noch:

- Relabel(5). Da $r_f(5,2) > 0$ und $dist(5) < dist(2)$ ist, folgt $dist(5) := dist(2) + 1 := 8$.

  Push(5,2) mit $\Delta = 3$. Dann ist $e(5) = 0$ und $e(2) = 3$
- Push(2,1) mit $\Delta = 3$. Dann ist $e(2) = 0$



Nun ist kein Knoten mehr aktiv, der Algorithmus terminiert. Und maximal Fluss ist 8.

Problem 2: Flüsse 2pt

Wir betrachten einen beliebigen, Maximalfluss f im Netzwerk $D = (V, E)$ und ein beliebiges Paar von Gegenkanten $\{(v, w), (w, v)\}$, es gelte zudem o.B.d.A. $f(v, w) \le f(w, v)$. Falls gilt $f(v, w) = 0$ so kann die Kante (v, w) offenbar ohne Folgen entfernt werden.

Sei nun also $f(v, w) > 0$ dann konstruiere das Netzwerk $D_{vw} = (V, E_{vw})$ mit $E_{vw} = E \setminus (v, w)$ und setze den neuen Fluss $f_{vw}(w, v)$ auf Kante (w, v) auf $f_{vw}(w, v) = f(w, v) - f(v, w)$. Für alle Kanten $e \in E_{vw} \setminus (v, w)$ bleibt der Fluss $f_{vw}(e)$ unverändert und somit sind die Flusserhaltungsbedingung in allen Knoten $z \in V \setminus v, w$ erhalten. Jedoch gilt die Flusserhaltungsbedingung auch für v, da $f(v, w)$ und $f(w, v)$ um den selben Betrag verringert wurden:

$$
\begin{aligned}
&\sum_{\{j|(v,j)\in E_{vw}\}} f_{vw}(v,j) - \sum_{\{j|(j,v)\in E_{vw}\}} f_{vw}(j,v) && (1)\\
&= \sum_{\{j|(v,j)\in E\}} f(v,j) \underbrace{-f(v,w)}_{\text{entferne } (v,w)} - \sum_{\{j|(j,v)\in E\}} f(j,v) \underbrace{+f(w,v) - (f(w,v) - f(v,w))}_{\text{setze neuen Fluss auf Kante } (w,v)} && (2)\\
&= \sum_{\{j|(v,j)\in E\}} f(v,j) - \sum_{\{j|(j,v)\in E\}} f(j,v) && (3)\\
&= 0 \quad (\text{da } f \text{ Fluss war}) && (4)
\end{aligned}
$$

Analoges gilt für den Knoten w. Somit ist f_{vw} ein Fluss. Es verbleibt zu zeigen, dass gilt $w(f_{vw}) = w(f)$. Doch da Gleichungen (1)-(3) insbesondere auch für den Knoten s gelten, ist der Wert des neuen Flusses gleich dem des alten Flusses (siehe Definition des Wertes im Skript).

Offenbar kann man mit diesem Verfahren iterativ (oder auch direkt) jedes Gegenkantenpaar auf eine einfache Kante reduzieren, ohne dass der Wert des Flusses verändert wird. Da keine Kantenkapazität erhöht wird, ist klar, dass der Wert des Maximalflusses durch dieses Verfahren nicht steigen kann, denn der Wert des Min-Cut kann durch das Entfernen von Kanten nicht steigen. Weiterhin gilt, dass die Kantenmenge nach erschöpfender Ausführung des Verfahrens maximal im Sinne der Aufgabe ist, da wir ausschließlich Gegenkantenpaare reduzieren. Eine Lösung des Flussproblems im resultierenden Netzwerk induziert direkt eine Lösung im ursprünglichen, nämlich eine Vereinfachung desjenigen Flusses, der für die Reduktion genutzt wurde.

Anmerkungen: Einfache Kanten mit Fluss Null könnten auch entfernt werden, dann würde man aber die geforderte Maximalität verlieren. Nach obigem Verfahren ist es durchaus möglich dass irgendein anderer maximaler Fluss, sollte es mehrere geben, nicht mehr möglich ist. Die obige Reduktion ist keine Reduktion im blichen Sinne, da die Problemstellung nicht erst in ein anderes Problem umgewandelt wird, sondern erst gelöst, und dann mit Hilfe der Lösung in ein anderes Problem umgewandelt wird. Offenbar ist die Reduktion jedoch polynomial. A priori kann man im Allgemeinen nicht ohne Kenntnis eines maximalen Flusses feststellen, welche Kante eines Gegenkantenpaares in einem Flussnetzwerk entfernt werden kann, ohne den Wert des Maximalflusses zu verringern. Das wesentliche Ergebnis besteht darin, dass Gegenkantenpaare niemals notwendig sind, um einen maximalen Fluss zu konstruieren.

Problem 3: Das Escape Problem 3pt

(a) Abbildung 1 zeit eine Ja-Instanz mit möglichst vielen Startknoten. Intuition: Es gibt 16 Knoten im Gitter aber nur 12 Randknoten, daher muss gelten $|S| \leq 12$. Abbildung 2 zeit eine Nein-Instanz mit möglichst wenigen Startknoten. Intuition: Ein innerer Gitterknoten hat immer mindestens 4 Wege zum Rand (geradeaus nach links/rechts/oben/unten), daher muss ein Knoten von mindestens 4 Knoten umbaut werden um jeden Weg zu einem Randpunkt zu blockieren.

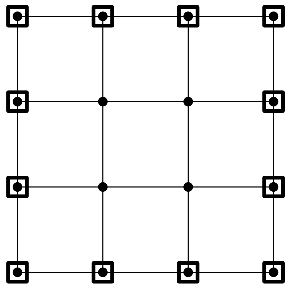


Abbildung 1: Große Ja-Instanz

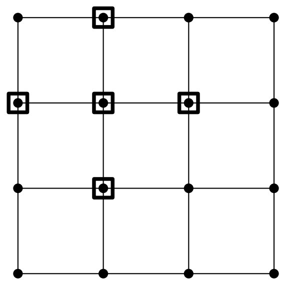


Abbildung 2: Kleine Nein-Instanz

(b) Abbildung 3 zeigt wie ein Knoten mit Knotenkapazität umgeformt werden muss, um ein solches Netzwerk auf ein normales Flussnetzwerk zurückzufhren. Die Umformung ist flussäquivalent, da

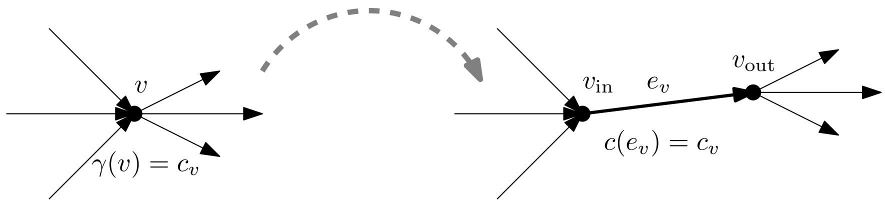


Abbildung 3: Knotenkapaziät wird flussäquivalent in Kantenkapazität umgeformt

durch die Kapazitätsbedingung der Kante e_v die Knotenkapazitätsbedingung $\gamma(v)$ realisiert wird und alle sonstigen Nachbarschaften und Kantenkapazitätsbedingungen beibehalten werden. Die Knotenkapazitätsbedingung beschränkt den Ausfluss eines Knotens v ebenso wie die Kapazität der Kante e_v. Für t_{out} ist analog der Einfluss beschränkt. Der Fluss im Rest des Netzwerkes bleibt durch diese Umwandlung somit unverändert. Es ist leicht zu sehen, dass jeder Fluss im Netzwerk vor der Umwandlung auch im Netzwerk nach der Umwandlung gültig ist, und umgekehrt.

Formal gilt für ein Netzwerk also: Konstruiere Netzwerk $N' = (D' = (V', E'); s'; t'; c')$ aus einem Netzwerk $N = (D = (V, E); s; t; c; \gamma)$, das Knotenkapazitäten enthält, wie folgt:

- Knoten: $V' = \bigcup_{v \in V} \{v_{\text{in}}, v_{\text{out}}\}$
- Quelle und Senke: $s' = s_{\text{in}}$, und $t = t_{\text{out}}$
- Kanten $E' = \{(v_{\text{out}}, w_{\text{in}}) \mid (v, w) \in E\} \quad \cup \quad \{(v_{\text{in}}, v_{\text{out}}) \mid v \in V\}$
- Kapazitäten: $c' : E' \to \mathbb{R}_0^+$ mit $c'(v_{\text{out}}, w_{\text{in}}) = c(v, w)$ und $c'(v_{\text{in}}, v_{\text{out}}) = \gamma(v)$

Korrektheit: Offenbar ist N' ein normales Flussnetzwerk von vergleichbarer Größe wie N ($|E'| = |E| + |V|$, $|V'| = 2|V|$). Da die Umwandlung eines jeden Knoten den Fluss im Rest des Netzwerkes unverändert lässt und die Knotenbedingungen erfüllt bleiben, gilt zudem, dass der maximale Fluss in N und in N' den gleichen Wert haben. Eine Lösung in N' kann kanonisch auf eine Lösung in N übersetzt werden.

(c) Zunächst konstruieren wir aus dem Gitter ein Flussnetzwerk mit Knotenkapazitäten $\gamma \equiv 1$, wie in

Abbildung 4 und wandeln dieses dann entsprechend Teilaufgabe (b) in ein normales Flussnetzwerk um. Die Intuition ist wie folgt:

Von der Quelle s verlaufen Kanten zu jedem möglichen Startknoten (Gefangene starten in Zellen), und nur von den Randknoten Kanten zur Senke t (Gefangene flüchten von Randknoten aus). Diese Kanten haben alle die Kapazität 1 da an jedem Gitterpunkt nur ein Startknoten liegt und an jedem Randknoten nur ein Weg enden darf.

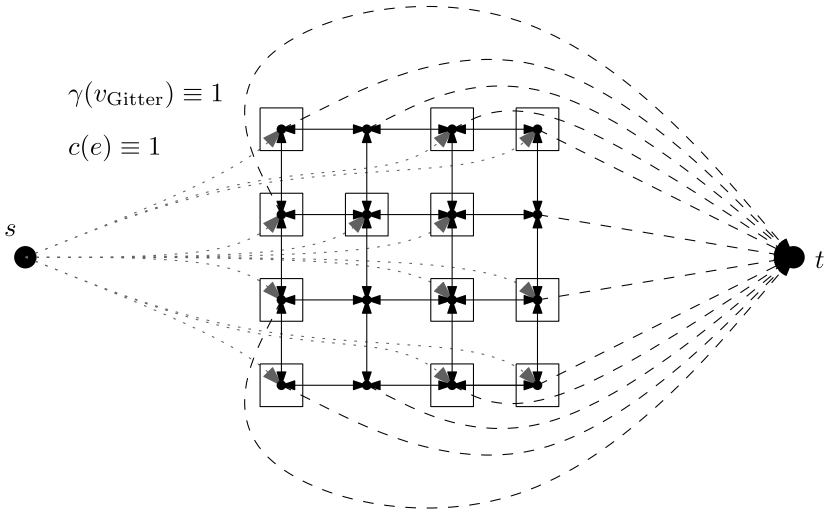


Abbildung 4: Das erweiterte Flussnetzwerk zu einer Instanz des Escape Problems (die Startknoten aus S sitzen in Kisten). Alle Startknoten werden von s gespeist, t ist Fluchtweg.

Die Gitterkanten sind bidirektional. Die Gitterknoten haben die Kapazität 1 da sie nur einmal genutzt werden dürfen. Die Forderung, dass Gitterkanten die Kapazität 1 haben ist zwar redundant (durch Knotenbedingungen schon erfüllt), macht die Situation aber verständlicher.

Ein maximaler Fluss in diesem Netzwerk entspricht dann direkt einer Menge von knotendisjunkten Wegen von Gitterknoten in S zu Randknoten. Hat der maximale Fluss den Wert $|S|$, so existiert genau für jeden Startknoten ein solcher Weg.

Wandeln wir nun dieses erweiterte Flussproblem entsprechend Teilaufgabe (b) in ein normales Flussnetzwerk um, so können wir dies mit bekannten Mitteln (etwa Goldberg-Tarjan) lösen. In Abbildung 5 ist dies illustriert.

Laufzeit:

- gegeben: Instanz der Größe n $\rightarrow n \times n$ Gitterpunkte

Transformation Flussproblem mit Knotenkapazität:

- $|V| = n^2 + 2$ Knoten
- $|E| = |S| < n^2$ Kanten von Masterquelle
- $+4 \cdot (n-1)$ Kanten zur Mastersenke
- $+2n(n-1) \cdot 2$ Kanten zw. Gitterpunkten
- $\rightarrow O(n^2)$ Knoten und $O(n^2)$ Kanten

Transformation Flussproblem ohne Knotenkapazität

- $|E'| = |E| + |V|$ Kanten $\in O(n^2)$
- $|V'| = 2|V|$ Knoten $\in O(n^2)$

Laufzeit Transformationen $\in O(n^2)$. Laufzeit max. Fluss wie folgt:

- Goldberg-Tarjan: $O(|V'|^2|E'|) = O(n^6)$.
- Goldberg-Tarjan-Highest-Label: $O(|V'|^2|E'|^{\frac{1}{2}}) = O(n^5)$.
- Goldberg-Tarjan-Excess-Scaling: $O(|E'||V'|+|V'|^2 \log C) \underset{\text{C konstant}}{=} O(|E'||V'|+|V'|^2) = O(n^4)$.
- Ford-Fulkerson: $O(|E'| \cdot |f^*|) \overset{\text{größte Ja-Instanz}}{\underset{\in O(4(n-1))}{=}} O(|E'| \cdot 4(n-1)) = O(n^3)$.

Insgesamt: Laufzeit $\in O(n^2 + n^3) = O(n^3)$.

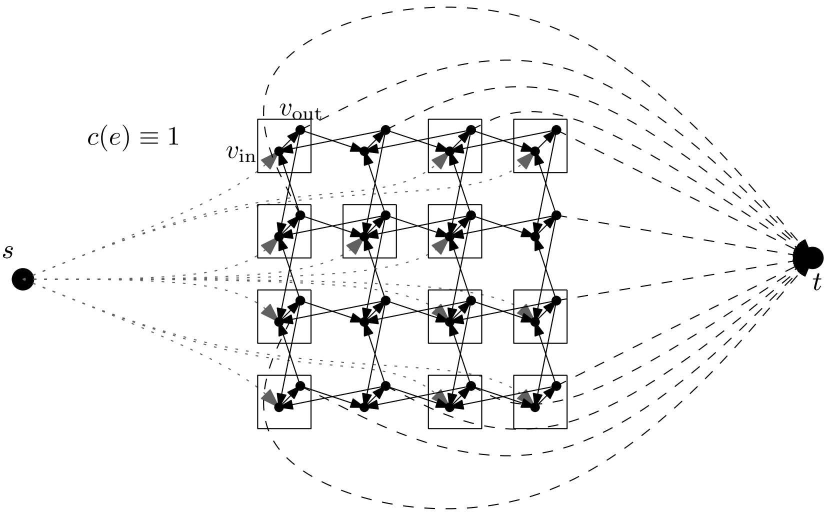


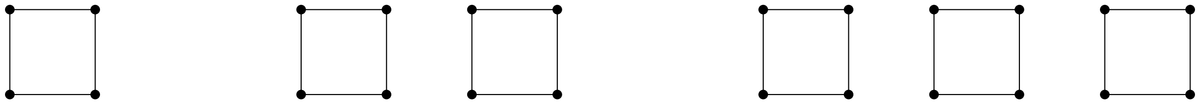
Abbildung 5: Das umgewandelte normale Flussnetzwerk. Aus Gründen der Übersicht wurden Quelle und Senke nicht verdoppelt, es liegen dort auch keine echten Knotenkapazitätsbedingungen vor.

Problem 4: Kreisbasen 2pt

(a) Wir konstruieren eine Familie von Graphen, parametrisiert ber $\alpha \in \mathbb{N}$, wie in Abbildung 6 angedeutet. Jeder Graph G_α^{exp} der Familie enthält α disjunkte einfache Kreise zu je vier Kanten:

- G_1^{exp} hat somit $|E| = 4$ Kanten und $|C| = 2$ allgemeine Kreise (die leere Menge und den einfachen Kreis).
- G_2^{exp} hat $|E| = 8$ Kanten und $|C| = 4$ allgemeine Kreise (die leere Menge, die beiden einfachen Kreise und den allgemeinen Kreis, der die Kanten beider einfachen Kreise enthält).
- G_3^{exp} hat $|E| = 12$ Kanten und $|C| = 8$ allgemeine Kreise (die leere Menge, drei einfachen Kreise, drei 2er-Kombination und den ,reis, der alle einfachen Kreisenenthält).
- Fr festes α beträgt damit die Anzahl Kanten $|E| = 4*\alpha$, die Anzahl allgemeiner Kreise dagegen beträgt $|C| = 2^\alpha = 2^{\frac{|E|}{4}} \in O(2^{|E|})$.

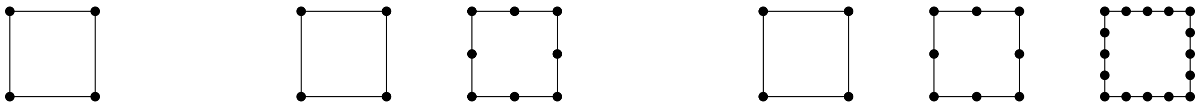
(b) Wir konstruieren eine Familie von Graphen, parametrisiert ber $\alpha \in \mathbb{N}$, wie in Abbildung 7 angedeutet. Jeder Graph G_α der Familie enthält α disjunkte einfache Kreise mit im Gegensatz zu *a)* exponentiell steigender Kantenzahl:



(a) $\alpha = 1$ (b) $\alpha = 2$ (c) $\alpha = 3$

Abbildung 6: die ersten drei Mitglieder der Familie G_α^{exp}

- G_1^{lin} hat somit $|E| = 4$ Kanten und $|C| = 2$ allgemeine Kreise (die leere Menge und den einfachen Kreis).
- G_2^{lin} hat $|E| = 12$ Kanten und $|C| = 4$ allgemeine Kreise (die leere Menge, die beiden einfachen Kreise und den allgemeinen Kreis, der die Kanten beider einfachen Kreise enthält).
- G_3^{lin} hat $|E| = 28$ Kanten und $|C| = 8$ allgemeine Kreise (die leere Menge, drei einfachen Kreise, drei 2er-Kombination und den Kreis, der alle einfachen Kreise enthält).
- Fr festes $\alpha > 1$ beträgt damit die Anzahl Kanten $|E| = 2^{\alpha+2} - 4$, die Anzahl allgemeiner Kreise dagegen beträgt $|C| = 2^\alpha = 2^{\log(|E|+4)-2} \in O(|E|)$.



(a) $\alpha = 1$ (b) $\alpha = 2$ (c) $\alpha = 3$

Abbildung 7: die ersten drei Mitglieder der Familie G_α^{lin}

(c) Geben Sie für untenstehenden Graphen (Kantengewichte stehen in Klammern) die fundamentale Kreisbasis zu dem hellgrau gestrichelten Baum an.

Zur Konstruktion der fundamentalen Kreisbasis:
Kante e_8 induziert C_1 : e_8 - e_4 - e_1 - e_5
Kante e_9 induziert C_2 : e_9 - e_5 - e_2 - e_6
Kante e_{10} induziert C_3 : e_{10} - e_6 - e_3 - e_7
Kante e_{11} induziert C_4 : e_{11} - e_4 - e_1 - e_2 - e_3 - e_7
Kante e_{12} induziert C_5 : e_{12} - e_3 - e_2 - e_1
Die Dimension einer Kreisbasis is $m - n + 1 = 5$. somit bilden diese fnf (linear unabhängigen) Kreise eine Kreisbasis.

(d) Drücken Sie den dunkelgrau gepunkteten Kreis als Linearkombination der Basiskreise aus.

Basisdarstellung des gestrichelten Kreises: e_{12} - e_7 - e_{11} - $e_4 = C_4 \oplus C_5$

(e) Zeigen oder wiederlegen Sie: Diese Fundamentalbasis ist eine minimale Kreisbasis.

Die Fundamentalbasis hat Gewicht $w(B) = \sum_{C \in B} w(C) = 44$

Wir zeigen mit Hilfe des Algorithmus von de Pina, dass jede minimale Kreisbasis Gewicht 39 hat. Somit ist die Fundamentalbasis *nicht* minimal:

- Initialisierung der $S_{1,j}$: $S_{1,1} = \{e_8\}, S_{1,2} = \{e_9\}, S_{1,3} = \{e_{10}\}, S_{1,4} = \{e_{11}\}, S_{1,5} = \{e_{12}\}$.
- $k = 1$:
  C_1 ist ein kürzester Kreis der eine ungerade Anzahl Kanten aus $S_{1,1}$ enthält, somit folgt:
  $C_1 = \{e_8, e_4, e_1, e_5\}$, und $w(C_1) = 8$.
  $S_{2,2} := S_{1,2}$, $S_{2,3} := S_{1,3}$, $S_{2,4} := S_{1,4}$, $S_{2,5} := S_{1,5}$
- $k = 2$:

C_2 ist ein kürzester Kreis der eine ungerade Anzahl Kanten aus $S_{2,2}$ enthält, somit folgt:
$C_2 = \{e_9, e_5, e_2, e_6\}$, und $w(C_2) = 10$.
$S_{3,3} := S_{2,3}$, $S_{3,4} := S_{2,4}$, $S_{3,5} := S_{2,5}$

- $k = 3$:
  C_3 ist ein kürzester Kreis der eine ungerade Anzahl Kanten aus $S_{3,3}$ enthätlt, somit folgt:
  $C_3 = \{e_{10}, e_6, e_3, e_7\}$, und $w(C_3) = 9$.
  $S_{4,4} := S_{3,4}$, $S_{4,5} := S_{3,5}$

- $k = 4$:
  C_4 ist ein kürzester Kreis der eine ungerade Anzahl Kanten aus $S_{4,4}$ enthält, somit folgt:
  $C_4 = \{e_{11}, e_4, e_{12}, e_7\}$, und $w(C_4) = 5$.
  $S_{5,5} := S_{4,4} \oplus S_{4,5} = e_{11}, e_{12}$

- $k = 5$:
  C_5 ist ein kürzester Kreis der eine ungerade Anzahl Kanten aus $S_{5,5}$ enthält, somit folgt:
  $C_5 = \{e_{12}, e_3, e_2, e_1\}$, und $w(C_5) = 7$.

Hinweis: Alternativ hätte man hier auch eine Kreisbasis geringeren Gewichts geschickt raten können, hätten allerdings beweisen mssen, dass diese tatsächlich die Basiseigenschaft erfllt, z.B. durch Bildung geeigneter Linearkombinationen.