

## 6 Mixed Max Cut in planaren Graphen und Via-Minimierung

Ein grundlegendes Problem besteht in der Berechnung eines *Schnittes* mit *minimalem* oder mit *maximalem Gewicht*. Die Komplexität dieses Problems ist wesentlich abhängig von der Gewichtsfunktion. Es gibt zahlreiche Anwendungen dieses Problems. Siehe auch Vorlesung „Algorithmentechnik“.

Wir werden einen polynomialen Algorithmus für die Berechnung eines Schnittes mit maximalem Gewicht in planaren Graphen mit beliebigen (positiven und negativen) Kantengewichten konstruieren. Darüber hinaus werden wir eine Anwendung dieses Algorithmus für das Via-Minimierungs-Problem, ein Problem aus dem „VLSI-Design“ (Entwurf hochintegrierter Schaltkreise) kennenlernen.

Eine Menge  $S \subseteq E$  heißt *Schnitt* von  $G = (V, E)$ , falls der durch  $E \setminus S$  induzierte Subgraph von  $G$  unzusammenhängend ist, d.h. in Graphen  $G_1 = (V_1, E_1)$ ,  $G_2 = (V_2, E_2)$  zerfällt, mit  $V_1 \cup V_2 = V$ ,  $V_1 \cap V_2 = \emptyset$ ,  $E_1 \cup E_2 = E \setminus S$ ,  $E_1 \cap E_2 = \emptyset$ , wobei alle Kanten aus  $S$  einen Endknoten in  $V_1$  und einen Endknoten in  $V_2$  haben.  $S$  *trennt* die Knoten  $u, v \in V$ , falls  $u$  und  $v$  in dem durch  $E \setminus S$  induzierten Subgraph (bezeichnet mit  $G - S$ ) in verschiedenen Zusammenhangskomponenten liegen. In einem Graph mit Kantengewichtsfunktion  $w : E \rightarrow K$  ist das *Gewicht* eines Schnittes  $S$  ist definiert als

$$w(S) := \sum_{e \in S} w(e) .$$

### MIN-CUT-PROBLEM

Gegeben sei ein Graph  $G = (V, E)$  mit einer Kantengewichtsfunktion  $w : E \rightarrow K$ , wobei  $K = \mathbb{R}^+$ . Finde einen Schnitt  $S \subseteq E$  mit  $w(S)$  minimal.

Das MIN-CUT-PROBLEM ist für beliebige Graphen in polynomialer Zeit lösbar, und zwar in Laufzeit  $\mathcal{O}(n \cdot m + n^2 \cdot \log n)$ . Siehe Vorlesung „Algorithmentechnik“.

### MAX-CUT-PROBLEM

Gegeben sei ein Graph  $G = (V, E)$  mit einer Kantengewichtsfunktion  $w : E \rightarrow K$ , wobei  $K = \mathbb{R}^+$ . Finde einen Schnitt  $S \subseteq E$  mit  $w(S)$  maximal.

Das MAX-CUT-PROBLEM ist für beliebige Graphen  $\mathcal{NP}$ -schwer.

## MIXED-MAX-CUT-PROBLEM

Gegeben sei ein Graph  $G = (V, E)$  mit einer Kantengewichtsfunktion  $w: E \rightarrow K$ , wobei  $K = \mathbb{R}$ . Finde einen Schnitt  $S \subseteq E$  mit  $w(S)$  maximal.

Das MIXED-MAX-CUT-PROBLEM ist für beliebige Graphen natürlich auch  $\mathcal{NP}$ -schwer. Sowohl MIN-CUT-PROBLEM und MAX-CUT-PROBLEM sind Spezialfälle des MIXED-MAX-CUT-PROBLEMS. Ersetze dazu beim MIN-CUT-PROBLEM  $w(e)$  durch  $-w(e)$ .

## 6.1 Mixed-Max-Cut in planaren Graphen

Wir werden nun einen Algorithmus für das MIXED-MAX-CUT-PROBLEM in planaren Graphen mit Laufzeit  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  angeben. Dieser basiert auf der Berechnung eines Matchings in planaren Graphen.

Wie zu erwarten, nutzt der Algorithmus für das MIXED-MAX-CUT-PROBLEM in planaren Graphen die Planarität ganz entscheidend aus, und zwar die Korrespondenz zwischen einem *Schnitt* in dem (eingebetteten) planaren Graphen  $G$  und einer *Menge von Kreisen* in dessen Dualgraph  $G^*$ .

Aus Lemma 2.9 folgt, dass das MIXED-MAX-CUT-PROBLEM in  $G = (V, E)$  äquivalent ist zu dem Problem, im Dualgraph  $G^* = (V^*, E^*)$  (bzgl. einer festen Einbettung von  $G$ ) eine nichtleere Menge von Kanten  $S^* \subseteq E^*$  zu finden, die kantendisjunkte Vereinigung von Kreisen ist, und für die  $w(S^*)$  maximal ist, wobei  $w(e^*) := w(e)$  für  $e^*$  Dualkante zu  $e$ . Wir benutzen folgenden *Satz von Euler*.

**Satz 6.1 (Satz von Euler).** *Für einen Graphen  $G = (V, E)$  sind äquivalent*

1.  $G$  ist Eulersch.
2.  $E$  ist kantendisjunkte Vereinigung einfacher Kreise.
3.  $d(v)$  ist gerade für alle  $v \in V$ .

Dabei heißt ein Graph  $G$  *Eulersch*, wenn jede Zusammenhangskomponente von  $G$  einen so genannten *Euler-Kreis* enthält, d.h. einen Kreis, der jede Kante genau einmal enthält. Zu einem Graphen  $G = (V, E)$  heißt eine Menge  $E' \subseteq E$  *gerade* genau dann, wenn in dem durch  $E'$  induzierten Subgraph von  $G$  jeder Knoten geraden Grad hat.

Das MIXED-MAX-CUT-PROBLEM in planaren Graphen ist also äquivalent zum MIXED-MAX-KREIS-PROBLEM.

## MIXED-MAX-KREIS-PROBLEM

Gegeben sei ein planarer Graph  $G = (V, E)$  mit einer Kantengewichtsfunktion  $w: E \rightarrow K$ , wobei  $K = \mathbb{R}$ . Finde eine nichtleere gerade Menge  $E' \subseteq E$  mit  $w(E')$  maximal.

Wir werden weiterhin die Äquivalenz des MIXED-MAX-KREIS-PROBLEM zu einem perfekten Matching minimalen Gewichts in einem geeignet definierten Graphen benutzen.

**Definition 6.2.** Ein Matchings  $M$  in einem Graphen mit einer geraden Anzahl  $n$  von Knoten heißt perfekt genau dann, wenn  $|M| = \frac{n}{2}$ .

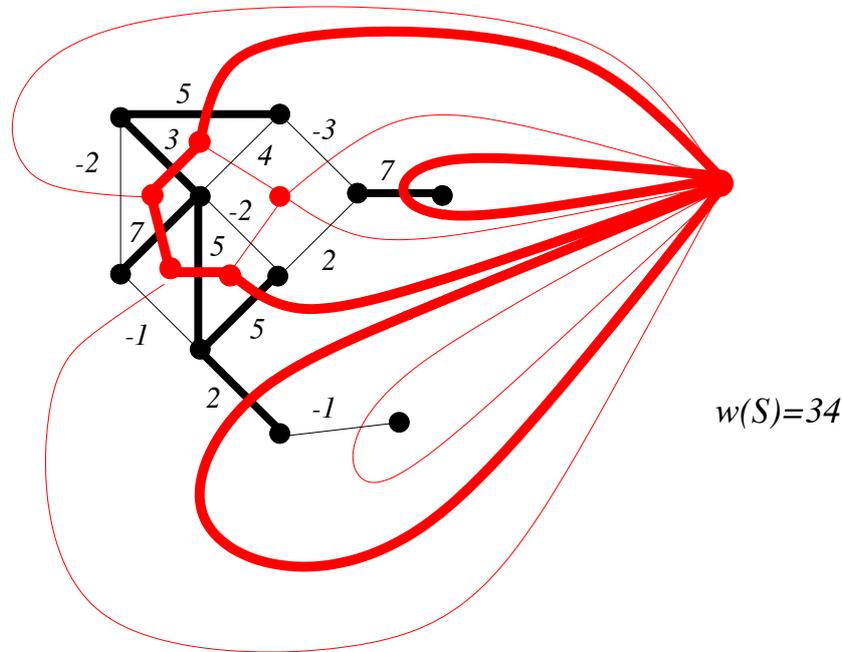


Abbildung 6.1: Illustration der Korrespondenz zwischen MIXED-MAX-CUT und MIXED-MAX-KREIS.

### Der Mixed-Max-Cut-Algorithmus von Shih, Wu & Kuo, 1990

Gegeben sei ein eingebetteter planarer Graph  $G = (V, E)$  mit Kantengewichtsfunktion  $w : E \rightarrow \mathbb{R}$ .

**Schritt 1:** Trianguliere  $G$  in  $\mathcal{O}(n)$  und ordne den hinzugefügten Kanten Gewicht 0 zu.

**Schritt 2:** Berechne in  $\mathcal{O}(n)$  den Dualgraph  $G^* = (V^*, E^*)$  zu der Triangulierung von  $G$ , wobei  $w(e^*) := w(e)$  mit  $e^*$  Dualkante zu  $e$ . Dann hat in  $G^*$  jeder Knoten Grad 3. Eine gerade Menge in  $G^*$  ist also eine knotendisjunkte Vereinigung einfacher Kreise in  $G^*$ .

**Schritt 3:** Konstruiere aus  $G^*$  in  $\mathcal{O}(n)$  einen Graph  $G' = (V', E')$  derart, dass ein perfektes Matching minimalen Gewichts in  $G'$  eine gerade Menge maximalen Gewichts in  $G^*$  induziert.

**Schritt 4:** Konstruiere in  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  ein perfektes Matching  $M$  minimalen Gewichts in  $G'$ .

**Schritt 5:** Falls  $M$  eine nichtleere gerade Menge in  $E^*$  induziert, gib den dazu dualen Schnitt in  $G$  aus. Ansonsten berechne in  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  aus  $M$  eine nichttriviale gerade Menge in  $G^*$  maximalen Gewichts.

**Ausführung von Schritt 3: Konstruktion von  $G' = (V', E')$**

$G^*$  ist 3-regulär (d.h., jeder Knoten hat Grad 3). Ersetze jeden Knoten  $v$  aus  $G^*$  durch einen Graph  $H_v$  mit 7 Knoten wie in Abbildung 6.2 und erhalte so  $G' = (V', E')$ . Die Gewichte der Kanten aus  $E^*$  werden dabei auf die entsprechenden Kanten aus  $E'$  übertragen und neue Kanten aus  $E'$  erhalten Gewicht 0. Wir unterscheiden nicht zwischen den Kanten vom Typ  $e_1, e_2$  und  $e_3$  in  $G^*$  und in  $G'$ .

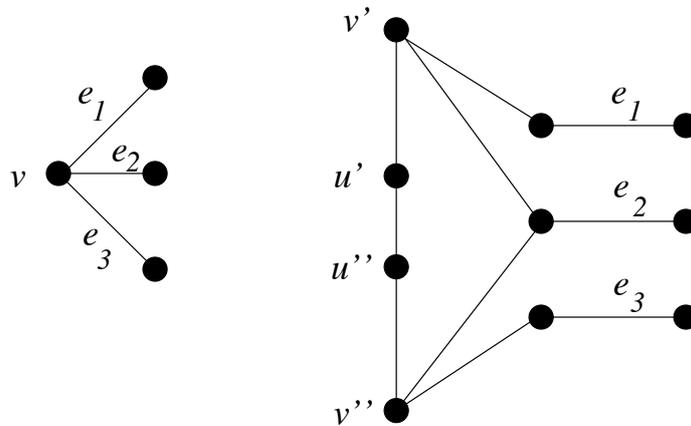


Abbildung 6.2: Ersetzung von  $v$  durch  $H_v$ .

*Beobachtung:* Da  $G^*$  3-regulär ist bzw. Dualgraph eines maximal planaren Graphen, ist  $|V^*|$  gerade, also auch  $|V'|$  gerade. Es existiert also in  $G'$  ein perfektes Matching.

**Lemma 6.3.** Sei  $G' = (V', E')$  entsprechend Abbildung 6.2 aus  $G^* = (V^*, E^*)$  konstruierter Graph.

- Falls  $M \subseteq E'$  ein perfektes Matching in  $G'$  ist, so ist die der Menge  $E' \setminus M$  entsprechende Menge  $M^* \subseteq E^*$  eine gerade Menge in  $G^*$ .
- Ist andererseits  $E_o^*$  eine gerade Menge in  $G^*$ , so induziert die der Menge  $M^* = E^* \setminus E_o^*$  entsprechende Teilmenge von  $E'$  ein perfektes Matching  $M$  in  $G'$ .

**Beweis.** „ $\implies$ “ Sei  $M$  ein perfektes Matching in  $G'$ . Betrachte für jeden Knoten  $v$  in  $G^*$  den entsprechenden Subgraphen  $H_v$  in  $G'$ .

**Fall 1:** Die Kante  $\{u', u''\}$  ist nicht in  $M$ . Dann sind die Kanten  $\{v', u'\}$  und  $\{u'', v''\}$  sowie  $e_1, e_2, e_3$  in  $M$ , also  $e_1, e_2, e_3$  nicht in der durch  $E \setminus M$  induzierten Menge  $M^* \subseteq E^*$ . Also ist  $d(v) = 0$  bzgl.  $M^*$  und damit  $M^*$  gerade Menge. Siehe Abbildung 6.3, links.

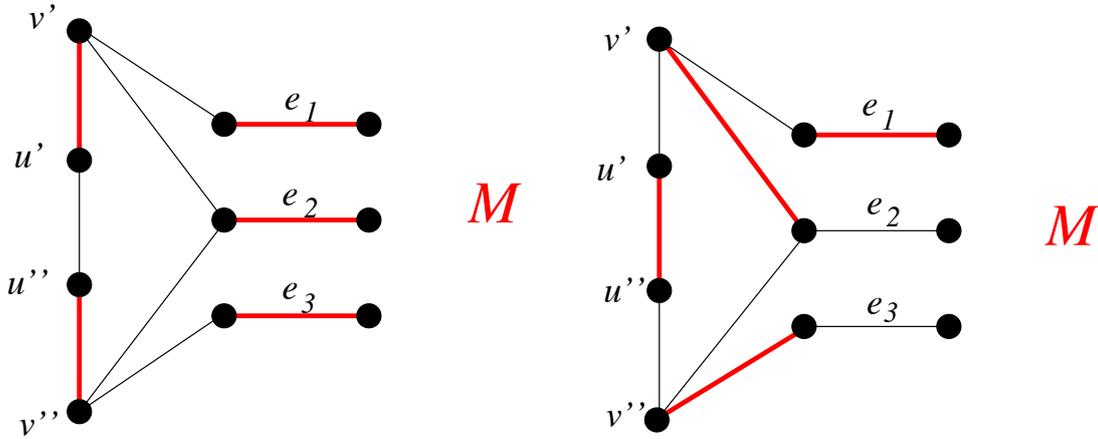


Abbildung 6.3: Illustration von Fall 1 (links) und Fall 2 (rechts).

**Fall 2:** Die Kante  $\{u', u''\}$  ist in  $M$ . Dann sind die Kanten  $\{v', u'\}$  und  $\{u'', v''\}$  nicht in  $M$ . Dementsprechend ist jeweils eine der anderen zu  $v'$  bzw.  $v''$  inzidenten Kanten in  $M$ , sowie genau eine der Kanten  $e_1, e_2, e_3$ . Also ist  $d(v) = 2$  bzgl. der durch  $E \setminus M$  induzierten Menge  $M^* \subseteq E^*$  und damit  $M^*$  gerade Menge. Siehe Abbildung 6.3, rechts.

„ $\Leftarrow$ “ Sei  $E_o^*$  eine gerade Menge in  $G^*$ . Dann haben alle Knoten in dem Subgraphen  $(V^*, E_o^*)$  von  $G^*$  entweder Grad 0 oder Grad 2.

**Fall 1:** Der Knoten  $v$  habe  $d(v) = 0$  bzgl.  $E_o^*$ . Dann enthalte  $M$  alle drei Kanten  $e_1, e_2, e_3$  und die Kanten  $\{v', u'\}$  und  $\{v'', u''\}$ . Siehe Abbildung 6.3, links.

**Fall 2:** Der Knoten  $v$  habe  $d(v) = 2$  bzgl.  $E_o^*$ , o.B.d.A.  $e_2, e_3 \in E_o^*$ . Dann enthalte  $M$  die Kante  $e_1$  sowie die Kante  $\{u', u''\}$  und die beiden Kanten inzident zu  $v'$  und  $v''$ , die zu den Kanten  $e_2$  bzw.  $e_3$  adjazent sind. Siehe Abbildung 6.3, rechts.

Dann ist  $M$  perfektes Matching in  $G'$  und die durch  $M$  induzierte Menge  $M^* \subseteq E^*$  erfüllt  $E^* \setminus M^* = E_o^*$ .  $\square$

**Folgerung 6.4.** Falls  $M \subseteq E'$  ein perfektes Matching minimalen Gewichts in  $G'$  ist, so ist die der Menge  $E' \setminus M$  entsprechende Menge  $M^* \subseteq E^*$  eine gerade Menge maximalen Gewichts in  $G^*$ . Ist andererseits  $E_o^*$  eine gerade Menge maximalen Gewichts in  $G^*$ , so induziert die der Menge  $M^* = E^* \setminus E_o^*$  entsprechende Teilmenge von  $E'$  ein perfektes Matching  $M$  minimalen Gewichts in  $G'$ .

**Beweis.** Es gilt  $w(E' \setminus M) = w(E^*) - w(E^* \cap M) = w(E^*) - w(M)$ , da alle  $e \in M$ , mit  $e \notin E^*$  Gewicht 0 haben.  $\square$

*Bemerkung:* Die durch  $M$  induzierte Menge  $M^*$  in  $G^*$  kann leer sein! Dazu später.

#### Ausführung von Schritt 4: Konstruktion eines perfekten Matchings minimalen Gewichts in $G'$ .

Zunächst kann ein perfektes Matching minimalen Gewichts mit einem Algorithmus zur Berechnung eines Matchings maximalen Gewichts folgendermaßen konstruiert werden.

*Beobachtung:*  $M$  ist ein perfektes Matching minimalen Gewichts in einem Graphen  $G = (V, E)$  mit Kantengewichten  $w : E \rightarrow \mathbb{R}$ , genau dann, wenn  $M$  ein perfektes Matching maximalen Gewichts in  $G = (V, E)$  mit Kantengewichten  $\bar{w} : E \rightarrow \mathbb{R}$ ,  $\bar{w}(e) := W - w(e)$ , wobei  $W$  geeignete Konstante.

Wir müssen nun noch die „Perfektheit“ von  $M$  bei der Berechnung erzwingen. Wähle dazu  $W$  geeignet. Zunächst gilt für ein perfektes Matching  $M$  in  $G$ , dass

$$\bar{w}(M) = \sum_{e \in M} \bar{w}(e) = \frac{n}{2} \cdot W - \sum_{e \in M} w(e) \geq \frac{n}{2} \cdot (W - w_{\max}),$$

wobei  $w_{\max} := \max_{e \in E} (w(e))$ . Für ein nicht-perfektes Matching  $M'$  gilt andererseits  $\bar{w}(M') \leq (\frac{n}{2} - 1) \cdot (W - w_{\min})$ , wobei  $w_{\min} := \min_{e \in E} (w(e))$ . Damit also  $\bar{w}(M) > \bar{w}(M')$  gilt für alle perfekten Matchings  $M$  und alle nicht-perfekten Matchings  $M'$  von  $G$ , reicht es aus,  $W$  so zu wählen, dass

$$\begin{aligned} \frac{n}{2}(W - w_{\max}) &> (\frac{n}{2} - 1)(W - w_{\min}), \\ \text{also } W &> \frac{n}{2}(w_{\max} - w_{\min}) + w_{\min} \end{aligned}$$

ist. Aus Kapitel 5 kennen wir einen Algorithmus mit Laufzeit  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  um in einem planaren Graphen ein Matching maximalen Gewichts, also auch ein perfektes Matching minimalen Gewichts zu bestimmen.

#### Ausführung von Schritt 5: Konstruktion des Schnitts.

Falls die durch  $E' \setminus M$  in  $E^*$  induzierte Menge  $M^*$  nicht leer ist, gib den entsprechenden dualen Schnitt in  $G$  aus.

Wir müssen nun noch den Fall behandeln, dass das berechnete perfekte Matching  $M$  minimalen Gewichts in  $G'$  die leere Menge in  $G^*$  induziert. Dazu berechnen wir in  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  aus  $M$  eine nichttriviale gerade Menge in  $G^*$  maximalen Gewichts. Diese hat dann offensichtlich negatives Gewicht!

Konstruiere zu jedem Knoten  $v \in V^*$  aus  $G'$  einen Graph  $G'_v$ , indem entsprechend Abbildung 6.4 in  $H_v$  die Kanten  $\{w', u'\}$  und  $\{w'', u''\}$  zugefügt werden. Die Kanten  $\{w', u'\}$  und  $\{w'', u''\}$  erhalten wieder Gewicht 0.

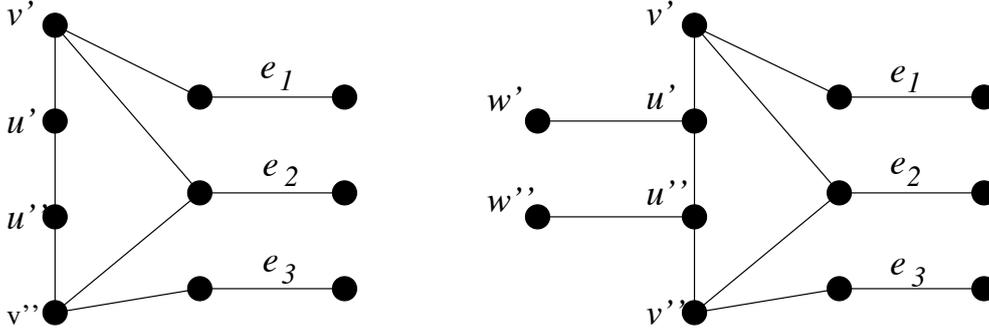


Abbildung 6.4: Konstruktion von  $G'_v$  aus  $G'$  zu ausgezeichnetem Knoten  $v \in V^*$ .

**Lemma 6.5.**  $M_v$  ist ein perfektes Matching minimalen Gewichts in  $G'_v$  genau dann, wenn die der Menge  $E'_v \setminus M_v$  entsprechende Menge in  $E^*$  eine gerade Kantenmenge maximalen Gewichts in  $G^*$  ist, die eine zu  $v$  inzidente Kante enthält.

**Beweis.** Ein perfektes Matching  $M_v$  in  $G'_v$  enthält immer  $\{w', u'\}$  und  $\{w'', u''\}$ . Ein solches Matching enthält dann genau eine der Kanten  $e_1, e_2, e_3$  zu  $v$ . Das Lemma folgt dann analog zu Lemma 6.3 und Folgerung 6.4.  $\square$

Ein perfektes Matching  $M_v$  minimalen Gewichts in  $G'_v$  erhält man in  $\mathcal{O}(n \log n)$  aus dem Matching  $M$  in  $G'$ . Wende dazu zweimal (für  $w'$  und  $w''$ ) den Algorithmus aus Kapitel 5 an. Betrachte nun für alle  $v \in V^*$  den Graph  $G'_v$ . Eine nichttriviale gerade Menge maximalen Gewichts in  $G^*$  wird dann durch die Menge  $M$  induziert, für die  $w(M) = \min_{v \in V^*} w(M_v)$ . Diese Menge  $M$  kann „direkt“ in  $\mathcal{O}(n^2 \cdot \log n)$  bestimmt werden.

Durch Anwenden des PLANAR-SEPARATOR-THEOREMS kommt man zu einem effizienteren Algorithmus mit Laufzeit  $\mathcal{O}(n^{\frac{3}{2}} \cdot \log n)$  wie folgt.

*Vorüberlegung:* Wenn für das in Schritt 4 berechnete perfekte Matching  $M$  minimalen Gewichts in  $G'$  gilt, dass die durch  $E' \setminus M$  induzierte Menge in  $G^*$  leer ist, so müssen alle Kreise in  $G^*$  negatives Gewicht haben. Die gesuchte nichttriviale gerade Menge in  $G^*$  besteht also aus einem einfachen Kreis negativen Gewichts, dessen Gewicht maximal ist unter allen Kreisen in  $G^*$ .

**Lemma 6.6.** In einem (nicht notwendigerweise einfachen) 3-regulären planaren Graphen  $G$ , der keinen positiven Kreis enthält, kann ein negativer einfacher Kreis maximalen Gewichts in  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  Zeit bestimmt werden.

**Beweis.** Wende folgenden Algorithmus an.

**Schritt 1:** Berechne eine Partition  $S, V_1, V_2$  in  $G$ , die die Bedingungen des PLANAR-SEPARATOR-THEOREMS erfüllt.

**Schritt 2:** Berechne rekursiv negative einfache Kreise maximalen Gewichts in den durch  $V_1$  und  $V_2$  induzierten Subgraphen  $G_1, G_2$  von  $G$ . Die Graphen  $G_1, G_2$  sind zwar nicht notwendigerweise 3-regulär, die 3-Regularität kann aber durch rekursives Entfernen von Grad-1-Knoten und Komprimieren von Grad-2-Knoten wieder hergestellt werden. (Diese Operationen erhalten Planarität und das Gewicht des schwersten Kreises.)

Für jedes  $v_i \in S$  berechne den negativen einfachen Kreis maximalen Gewichts in  $G$ , der  $v_i$  enthält, wie folgt:

Konstruiere zu  $G$  den Graphen  $G'$  gemäß Schritt 3 des Mixed-Max-Cut-Algorithmus. Für jeden Knoten  $v_i \in S$  erweitere  $G'$  zu  $G'_{v_i}$ , indem  $v_i$  durch den durch  $\{u', u'', v', v'', w', w''\}$  induzierten Subgraphen (vgl. Abb. 6.4, rechts) ersetzt wird (jeder dieser Graphen enthält also die Knoten  $w'$  und  $w''$  genau einmal), und bestimme in  $\mathcal{O}(n \log n)$  ein perfektes Matching minimalen Gewichts von  $G'_{v_i}$ ; berechne den dazu korrespondierenden negativen einfachen Kreis maximalen Gewichts in  $G$ .

**Schritt 3:** Gib den Kreis maximalen Gewichts unter allen konstruierten Kreisen aus. Dieser ist der gewünschte Kreis in  $G$ , da jeder einfache Kreis in  $G$  entweder ganz in  $G_1$  oder ganz in  $G_2$  liegt oder (mindestens) ein  $v_i \in S$  enthält.

Die Gesamtlaufzeit  $t(n)$  ist gegeben durch

$$\begin{aligned} t(n_0) &= c_0 \\ t(n) &= t(c_1 \cdot n) + t(c_2 \cdot n) + c_3 \cdot \sqrt{n} \cdot n \cdot \log n, \end{aligned}$$

wobei  $c_0, c_1, c_2, c_3$  konstant,  $c_1, c_2 \leq \frac{2}{3}$  und  $c_1 + c_2 < 1$ . Damit ist  $t(n) \in \mathcal{O}(n^{\frac{3}{2}} \log n)$ .  $\square$

Damit haben wir insgesamt folgenden Satz bewiesen.

**Satz 6.7.** *In einem planaren Graphen  $G = (V, E)$  mit Kantengewichtsfunktion  $w : E \rightarrow \mathbb{R}$  kann in  $\mathcal{O}(n^{\frac{3}{2}} \log n)$  ein Schnitt  $S \subseteq E$  konstruiert werden, mit  $w(S)$  maximal.*

## 6.2 Das Via-Minimierungs-Problem

Eine von vielen interessanten Anwendungen des MIXED-MAX-CUT-PROBLEMS tritt beim Entwurf hochintegrierter Schaltungen auf. Man möchte eine Schaltung möglichst kostengünstig auf einem Chip realisieren. Ein Schritt in dem entsprechenden Entwurfsprozess besteht darin, ein Layout der Schaltung so innerhalb mehrerer Lagen zu realisieren, dass die Anzahl der Lagenwechsel klein ist. Als Basis der Realisierung einer Schaltung wird üblicherweise ein orthogonales Gitter angenommen.

Eine *Schaltung* bestehe aus *Modulen*, auf deren Rändern *Terminale* liegen und *Drähten*, die jeweils vorgegebene Terminale verbinden. Ein *Layout*  $L$  ist dann eine Einbettung der Schaltung in ein orthogonales Gitter, bei der Drähte als kantendisjunkte Verbindungen (im allgemeinen *Steiner-Bäume*) entlang Gitterlinien geführt werden. Wir werden uns hier auf den Fall beschränken, dass jeder Draht genau zwei Terminale verbindet, d.h. die Drähte als kantendisjunkte *Wege* eingebettet werden können.

Jede *Lage* ist dann eine Kopie des orthogonalen Gitters, und eine zulässige *Lagenzuweisung* besteht in einer knotendisjunkten Zuordnung der eingebetteten Drahtstücke zu Lagen, d.h. keine zwei Drähte berühren sich in derselben Lage. Lagenwechsel, so genannte *Vias*, sind nur an Gitterpunkten erlaubt. Wenn es zu einem Layout eine zulässige Lagenzuweisung in zwei Lagen gibt, so nennt man das Layout auch *in zwei Lagen realisierbar*. Siehe Abbildungen 6.5 und 6.6.

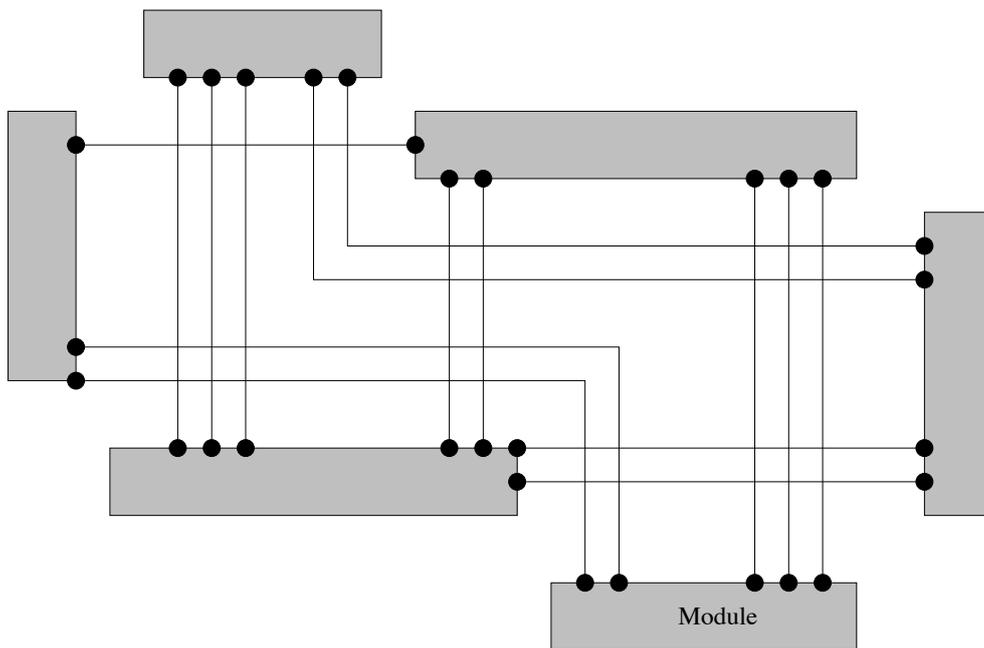


Abbildung 6.5: Ein in zwei Lagen realisierbares Layout.

#### VIA-MIMIMIERUNGS-PROBLEM

Gegeben sei ein in zwei Lagen realisierbares Layout  $L$ . Finde eine Realisierung von  $L$  in zwei Lagen mit minimaler Anzahl an Vias.

*Bemerkung:* Ein Layout, bei dem sich verschiedene Drähte in Gitterpunkten kreuzen dürfen, aber nicht an dem selben Gitterpunkt gegeneinander abknicken dürfen, können immer leicht in zwei Lagen realisiert werden. Dazu ordnet man einfach alle vertikalen Drahtstücke der einen und alle horizontalen Drahtstücke der anderen Lage zu. Solche Layouts werden *Manhattan-Layout* genannt. Eine solche Realisierung in zwei Lagen für das Layout aus Abbildung 6.5 würde vier Vias benötigen. In Abbildung 6.6 ist eine

## 6 Mixed Max Cut und Via-Minimierung

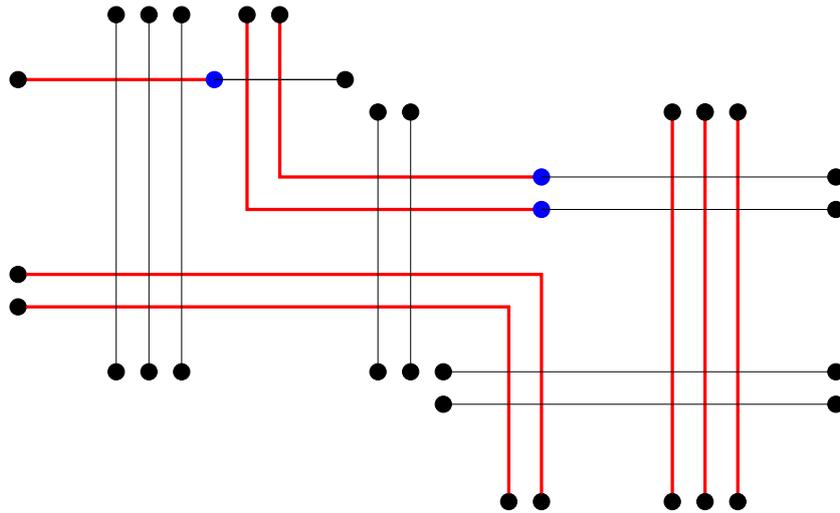


Abbildung 6.6: Realisierung des Layouts aus Abbildung 6.5 in zwei Lagen mit 3 Vias.

Realisierung dieses Layouts in zwei Lagen mit 3 Vias gezeigt. Ist für das Layout in Abbildung 6.5 eine Realisierung in zwei Lagen mit weniger als 3 Vias möglich?

*Vorüberlegung:* Eine Realisierung eines Layouts in zwei Lagen entspricht einer Zweifärbung der Drahtstücke. Die Anzahl der Vias entspricht der Gesamtzahl der Farbwechsel von Drähten. Entsprechend werden wir das VIA-MINIMIERUNGSPROBLEM als ein Graphenfärbungsproblem modellieren. Zunächst müssen wir die entscheidenden Charakteristika des Problems, die einerseits in den *Konflikten* zwischen Drahtstücken und andererseits in der geeigneten Wahl von *Lagenwechseln durch Vias* bestehen, durch einen Graph ausdrücken. Dazu definieren wir den *Konfliktgraph* zu einem Layout.

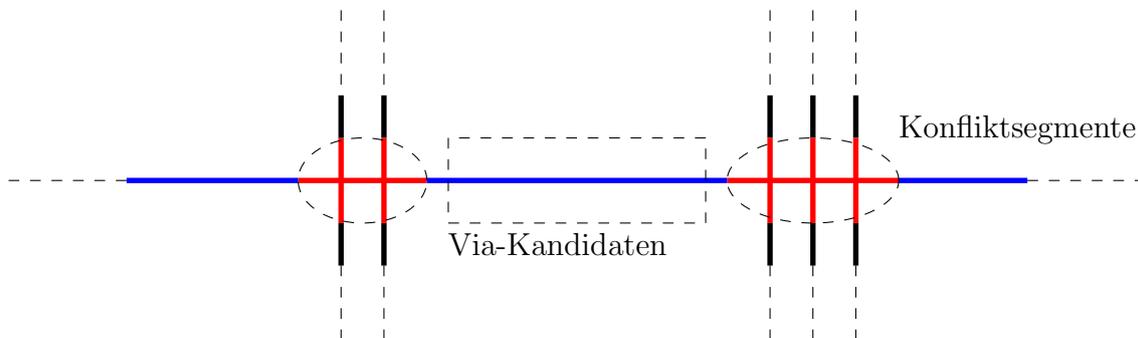


Abbildung 6.7: Ausschnitt eines Layouts und dessen Aufteilung in *Konflikt-Segmente* und *Via-Kandidaten*.

Die Drähte eines Layouts können in zwei Typen von Drahtstücken aufgeteilt werden, in *Konflikt-Segmente* und in *Via-Kandidaten*. Siehe Abbildung 6.7.

- *Konflikt-Segmente* sind Drahtstücke, die in allen Gitterpunkten andere Drähte berühren.

- Die restlichen Drahtstücke sind *Via-Kandidaten*, d.h. maximale Drahtstücke, die über mindestens einen Gitterpunkt gehen, über den kein anderer Draht geht. Dies sind gerade die Drahtstücke, auf denen Vias plaziert werden können.

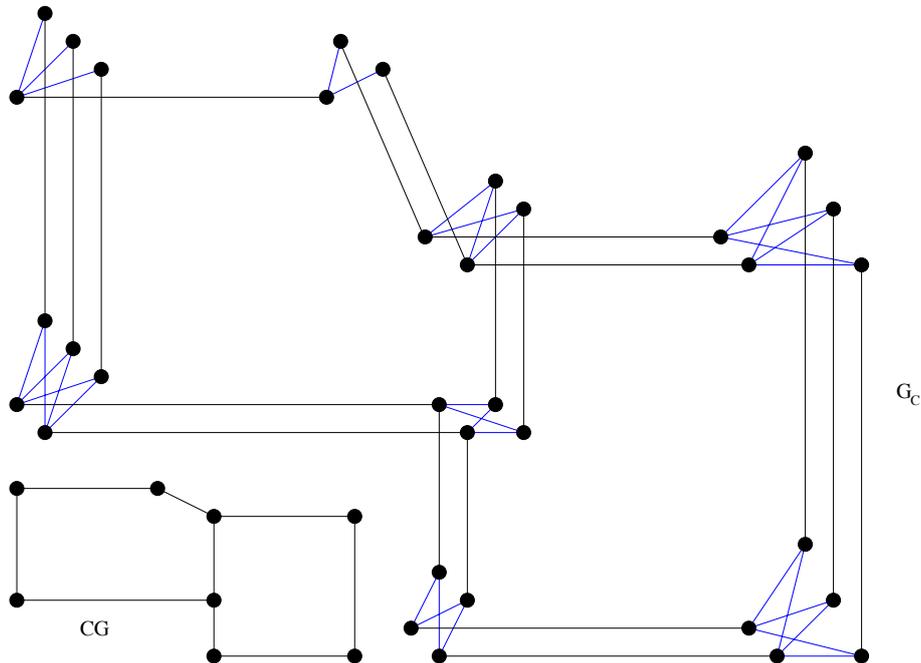


Abbildung 6.8: Der Konfliktgraph zum Layout aus Abbildung 6.5 und der zugehörige Clustergraph.

Zu einem in zwei Lagen realisierbaren Layout  $L$  definiere den *Konfliktgraph*  $G_c(L) = (V_c, E_c)$  wie folgt.

- $V_c$  entspricht der Menge aller Konflikt-Segmente
- $E_c$  enthält zwei Typen von Kanten
  - $\{u, v\} \in E_c$  für Knoten  $u, v \in V_c$ , welche Konflikt-Segmenten entsprechen, die sich in einem Gitterpunkt berühren, genannt *Konfliktkanten*
  - $\{u, v\} \in E_c$  für Knoten  $u, v \in V_c$ , welche Konflikt-Segmenten entsprechen, die inzident zu dem selben Via-Kandidaten sind.

Die Subgraphen von  $G_c$ , die durch Konfliktkanten induziert sind, d.h. Mengen von Konfliktsegmenten, die sich gegenseitig berühren, können zu *Konfliktclustern* zusammengefasst werden. Dadurch wird ein bewerteter *Clustergraph*  $CG := (CV, CE)$  induziert, mit Kantenbewertung  $c : CE \rightarrow \mathbb{N}$ .

- $CV$  entspricht der Menge der Konfliktcluster,
- $\{a, b\} \in CE$ , falls in dem Konfliktcluster zu  $a$  und dem Konfliktcluster zu  $b$  Konfliktsegmente existieren, die durch denselben Via-Kandidaten verbunden sind,

## 6 Mixed Max Cut und Via-Minimierung



Abbildung 6.9: Verschwindende Vias (a) und Kanten, die sich bezüglich Vias unterschiedlich verhalten, im Clustergraph aber auf eine Kante abgebildet werden (b).

–  $c(e) := \#$  der Via-Kandidaten, die  $e$  entsprechen für  $e \in CE$ .

Abbildung 6.8 zeigt den Konfliktgraph und zugehörigen Clustergraph zum Layout aus Abbildung 6.5. Offensichtlich ist CG immer planar. Achtung, es kann passieren, dass Viakandidaten beim Übergang zum Clustergraphen in Clustern “verschwinden”, siehe Abbildung 6.9(a). Da Cluster am durch die Lagezuweisung eines einzelnen Knoten vollständig bestimmt sind, sind dies gerade die Via-Kandidaten, die in jedem gültigen Layout einen Via enthalten bzw. in keinem gültigen Layout einen Via enthalten. Zudem müssen sich nicht alle Via-Kandidaten, die im Clustergraphen auf dieselbe Kante abgebildet werden gleich verhalten. Im Beispiel aus Abbildung 6.9(b) ist in jedem gültigen Layout auf genau einem der beiden Via-Kandidaten ein Via vorhanden. Dennoch werden beide auf dieselbe Kante im Clustergraphen abgebildet.

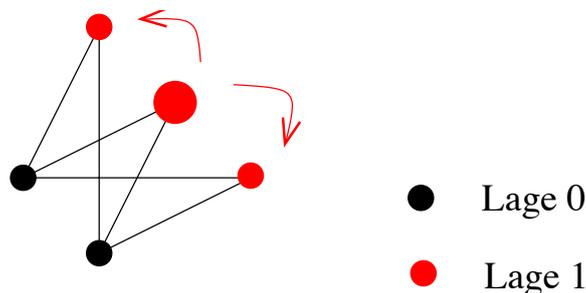


Abbildung 6.10: Festlegung der Lagen aller Konfliktsegmente eines Konflikt-Clusters durch Festlegung des Repräsentanten.

In einer Realisierung von  $L$  in zwei Lagen ist die Lage aller Konfliktsegmente eines Konflikt-Clusters durch die Lage eines einzigen Konfliktsegments dieses Clusters festgelegt. Man kann also einen beliebigen Knoten eines jeden Konflikt-Cluster als *Repräsentanten* der Lagenzuweisung wählen. Im folgenden gehen wir von einer fest gewählten Menge von Repräsentanten aus.

Eine Realisierung von  $L$  in zwei Lagen entspricht zunächst einer Zweifärbung der Knoten aus  $G_c$ , so dass Knoten, die durch eine Konfliktkante verbunden sind, verschiedene Farben haben. Dies entspricht einer Färbung der Knoten von CG mit zwei Farben, wobei

## 6 Mixed Max Cut und Via-Minimierung

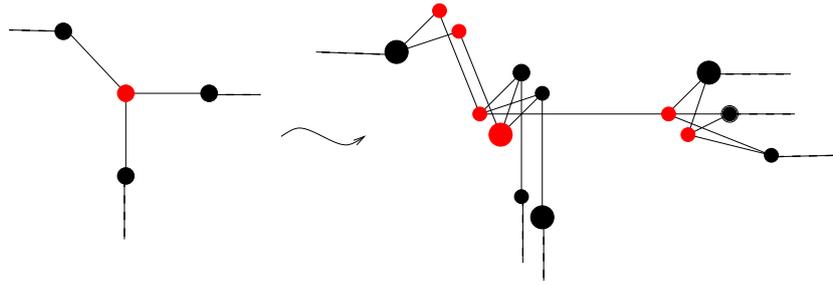


Abbildung 6.11: Festlegung der Lagen entsprechend einer Zweifärbung im Konfliktgraph.

adjazente Knoten die gleiche Farbe haben können. Eine „echte“ Zweifärbung von  $G_C$  würde dann einer Realisierung von  $L$  in zwei Lagen entsprechen ohne Vias. Bei geeigneter Wahl der Repräsentanten entspricht dies auch einer echten Zweifärbung von  $CG$ . Eine „echte“ Zweifärbung eines Graphen existiert genau dann, wenn der Graph keine Kreise ungerader Länge enthält. Solche Graphen heißen *bipartit*. Es ist also leicht zu entscheiden, ob ein Layout ohne Via in zwei Lagen realisierbar ist.

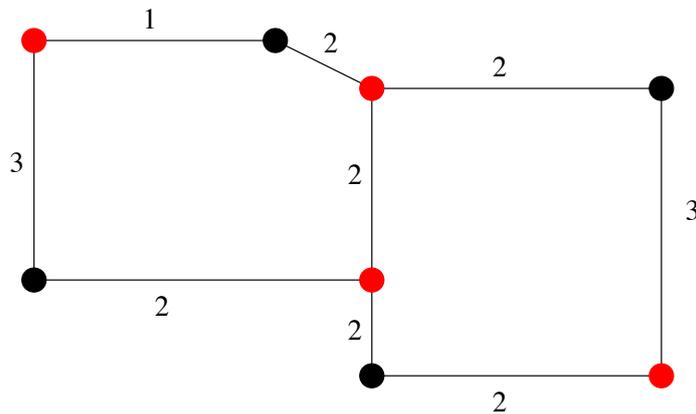


Abbildung 6.12: Zweifärbung des Cluster-Graphen.

Für eine beliebige, aber feste Realisierung von  $L$  in zwei Lagen, d.h. eine Färbung der Knoten von  $CG$  mit zwei Farben, sei  $v(e)$  die *Anzahl der Vias* auf Via-Kandidaten, die  $e$  entsprechen. Da die sich die Färbung der Repräsentanten eindeutig auf den Konfliktgraphen  $G_C$  erweitern lassen kann, kann die Funktion  $v(e)$  leicht bestimmt werden. Wir definieren nun für jede Kante  $e$  von  $CG$  die *Via-Reduktion*  $v_{\text{red}}(e)$ , die angibt, um welchen Wert sich die Anzahl der Vias durch Vertauschen der Lagenzuweisung für ein Konflikt-Cluster, das mit einem zu  $e$  gehörenden Via-Kandidaten inzident ist, verringert. Also

$$v_{\text{red}}(e) := v(e) - (c(e) - v(e)) = 2v(e) - c(e).$$

## 6 Mixed Max Cut und Via-Minimierung

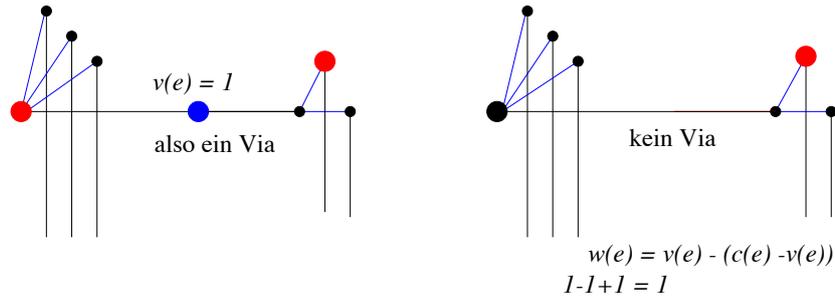


Abbildung 6.13: Reduktion um ein Via bzgl.  $v$ .

Sei  $X \subseteq CV$ , dann ist die *Via-Reduktion für  $X$*  bzgl.  $v$  entsprechend definiert als

$$v_{\text{red}}(X) := \sum_{\substack{e=\{x,y\} \in CE \\ x \in X, y \notin X}} v_{\text{red}}(e)$$

Das heißt, dass  $v_{\text{red}}(X)$  die Via-Reduktion bzgl. der zu  $v$  gehörigen Realisierung in zwei Lagen bei Vertauschen der Lagen für alle Konflikt-Cluster, die Knoten aus  $X$  entsprechen, ist.

Jede Realisierung eines Layouts  $L$  in zwei Lagen kann aus einer beliebigen Realisierung von  $L$  in zwei Lagen durch Vertauschen der Lagen für eine geeignete Menge von Konflikt-Clustern erreicht werden. Damit ist das **VIA-MINIMIERUNGS-PROBLEM** äquivalent zu folgendem Maximierungsproblem.

### MAX-VIA-REDUKTION

Gegeben der planare Clustergraph  $CG = (CV, CE)$  mit Kantenbewertung  $c$  zu einem Layout und eine (nicht notwendig echte) Färbung der Knoten von  $CG$  mit zwei Farben. Sei  $v : CE \rightarrow \mathbb{Z}$  die zugehörige Funktion der Via-Anzahl und  $v_{\text{red}}$  die entsprechende Funktion der Via-Reduktion.

Finde  $X \subseteq CV$ , so dass  $v_{\text{red}}(X)$  maximal ist.

Da jedes  $X \subseteq CV$  einen Schnitt in  $CG$  induziert, und  $v_{\text{red}}(X)$  gerade das Gewicht dieses Schnittes bzgl. der Kantengewichtsfunktion  $v_{\text{red}} : CE \rightarrow \mathbb{Z}$  ist, entspricht dieses Problem gerade dem *Mixed-Max-Cut-Problem*.

Startend bei einer beliebigen Wahl der Repräsentanten und einer beliebigen Färbung, besteht unser Problem also in der maximalen Via-Reduktion, also der Wahl einer Knotenmenge  $X \subseteq CV$  so, dass der zugehörige Schnitt maximales Gewicht hat.

*Beachte:*  $v_{\text{red}}$  kann auch negative Werte haben.

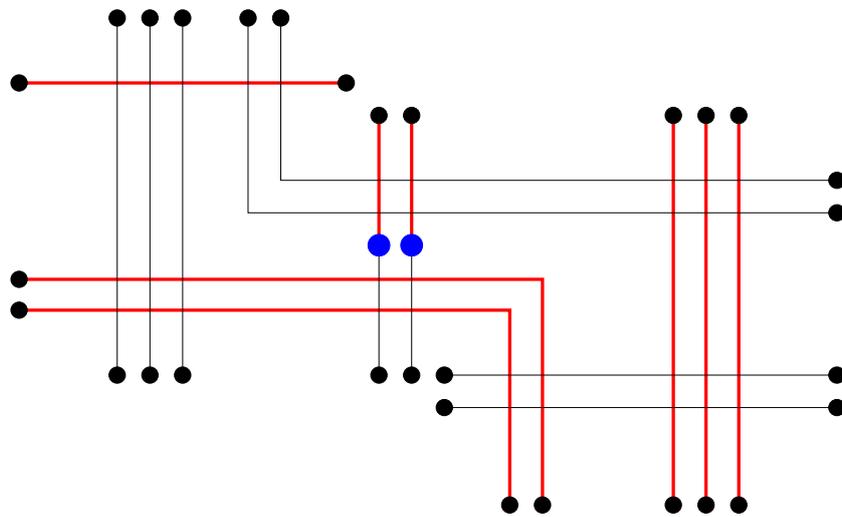


Abbildung 6.14: Realisierung des Layouts aus Abbildung 6.5 in zwei Lagen mit nur 2 Vias.