

Parameterisierte Algorithmen

SoSe 2013 in Trier

Henning Fernau

fernau@uni-trier.de

Parameterisierte Algorithmen

Gesamtübersicht

- Einführung
- Grundbegriffe
- **Problemkerne**
- Suchbäume
- Graphparameter
- Weitere Methoden
- Komplexitätstheorie—parameterisiert

Wie erhalte ich Kernreduktionen?

- Fokus auf Extremen (z. B. kleingradige / großgradige Knoten)
- Verwendung bekannter (nicht-trivialer) mathematischer Aussagen
- Dafür häufig wichtig: Betrachtung annotierter Probleme
- Speziell: Einsatz von Katalysatoren
- Verallgemeinerung bekannter einfacher Regeln

Eine weitere Reduktionsidee für VC: Kronen

Es sei $G = (V, E)$ ein Graph. Eine unabhängige Menge I von G heißt **Krone** gdw. für alle nicht-leere Mengen $U \subseteq N(I)$: $|U| \leq |N(U) \cap I|$.

$H := N(I)$ wird in I (gepaart) gematched. (H heißt auch **Kopf** der Krone.)

Kurzer Exkurs: Der Graph $G[I \cup H]$ erinnert an den **Heiratssatz von Hall**):

Sei $G = (V_1, V_2; E)$ ein bipartiter Graph mit $|V_1| \leq |V_2|$.

G besitzt ein Matching der Größe $|V_1|$ gdw. für alle $U \subseteq V_1$ ist $|U| \leq |N(U)|$.

Reduktionsregel 1 *Ist $(G = (V, E), k)$ eine VC Instanz und $I \subseteq V$ eine Krone mit Kopf $H = N(I)$, so reduziere zu $(G - (I \cup H), k - |H|)$.*

Problem: Wie findet man (große) Kronen?

Standarddefinition für Kronen

Eine **Kronenzerlegung** eines Graphen $G = (V, E)$ ist ein Tripel (I, H, X) mit $I \cup H \cup X = V$, $I \cap H = \emptyset$, $I \cap X = \emptyset$, $H \cap X = \emptyset$, sodass

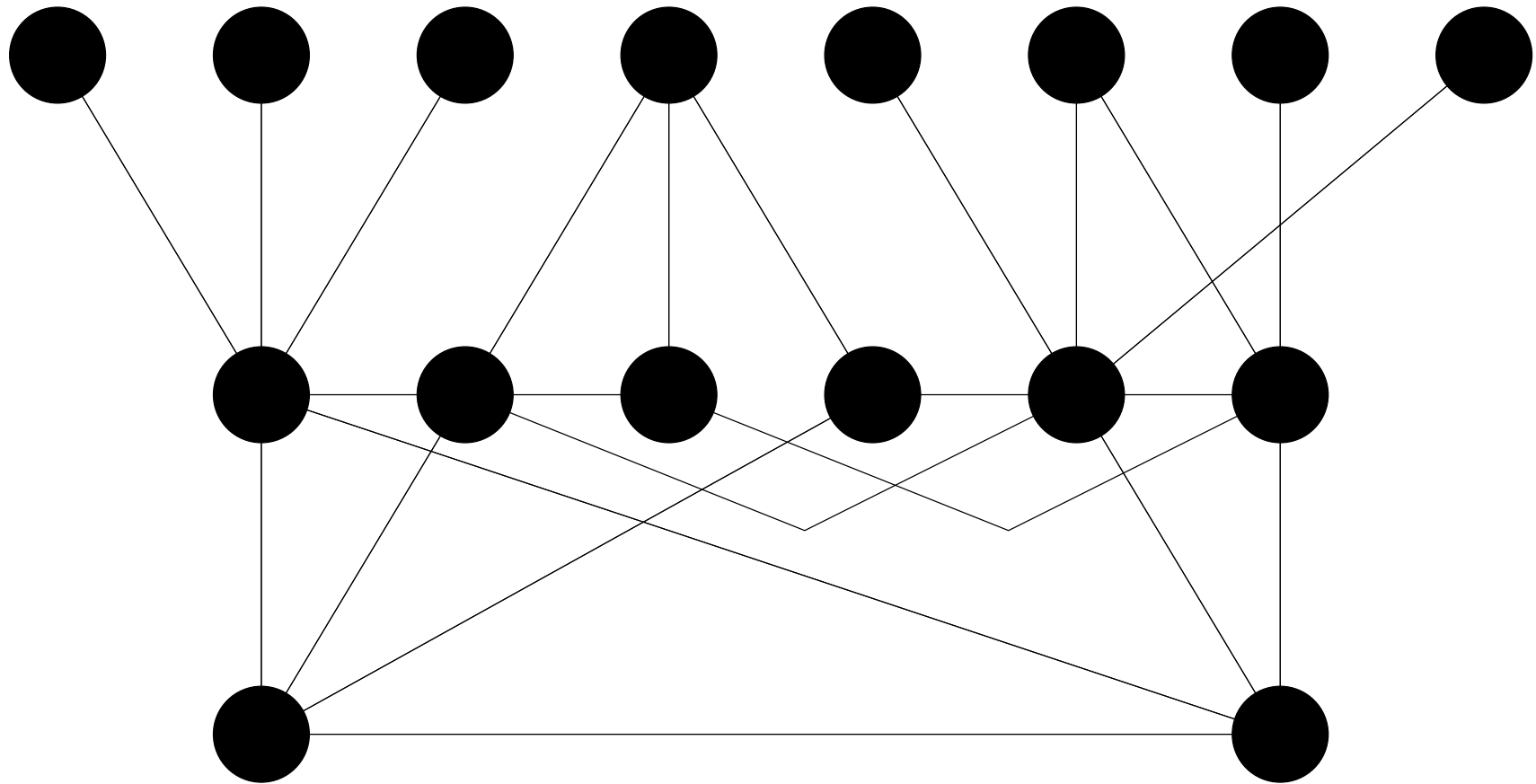
1. I ist eine unabhängige Menge in G .
2. H ist ein Separator (zwischen I und X).
3. H wird in I gepaart durch eine injektive Abbildung $\mu : H \rightarrow I$.

Dann ist I eine Krone in der Definition der vorigen Folie.

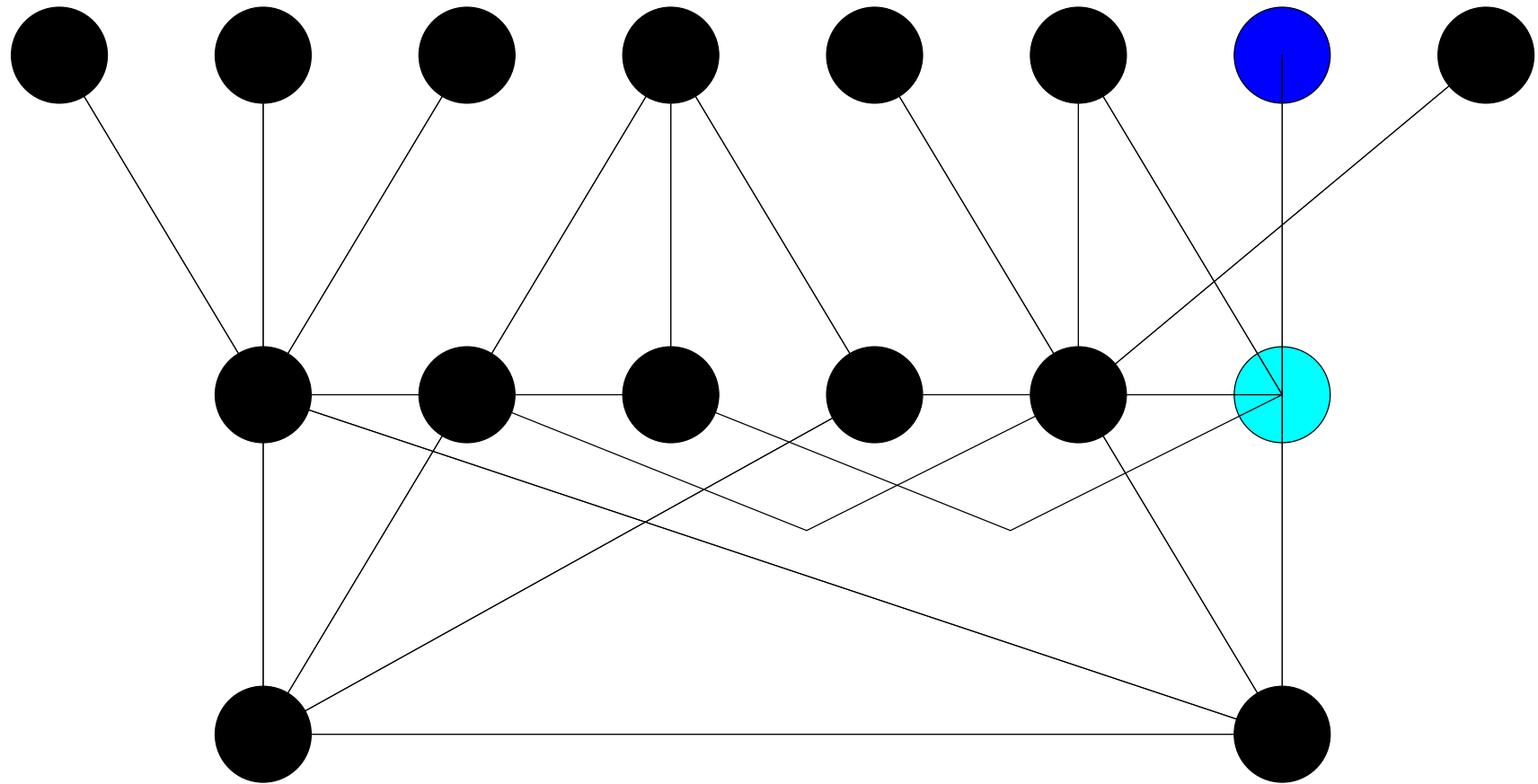
Die Kronenreduktionsregel sagt aus:

Ist (I, H, X) Kronenzerlegung von G , so kann man sich zur Berechnung (kleinster!) Knotenüberdeckungen auf die Betrachtung von $G[X]$ zurückziehen.

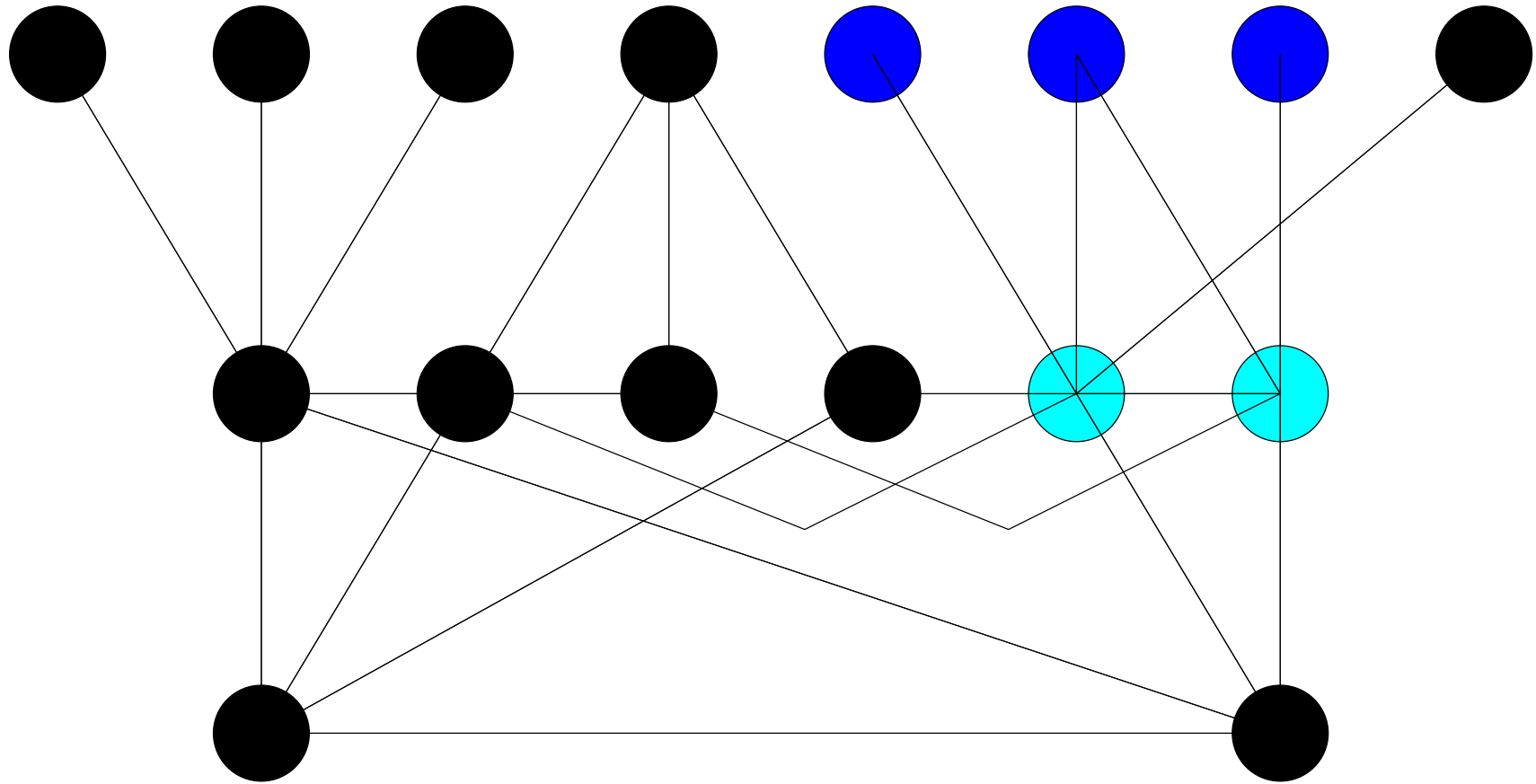
Die Kronen an einem Beispiel



Eine kleine blaue Krone mit Kopf



Eine größere blaue Krone mit Kopf



Lemma 1 *Alg. 2 kombiniert mit Regel 1, angesetzt auf (G, k) , liefert entweder korrekt NEIN oder liefert eine reduzierte Instanz (G', k') von VC mit $|V(G')| \leq 3k$.*

Beobachte hierzu:

- Finden wir ein Matching M (d.h., M_1 oder M_2) in G mit $|M| > k$, so ist (G, k) eine \times -Instanz wegen des Lemmas von König und Egerváry.
- Enthalten weder M_1 noch M_2 mehr als k Kanten, so hat $G[M_1 \cup M_2]$ höchstens $3k$ Knoten.

Algorithm 1 An efficient algorithm to find a crown in a graph, called GETCROWN

Input(s): an graph G

Output(s): a crown $I \subseteq V(G)$

Greedy find a **maximal** matching M_1 of G .

Let O (outsiders) be the set of vertices from G that are not in the matching.

if $O \neq \emptyset$ **then**

$I' \leftarrow \emptyset$.

Find a **maximum** matching M_2 in the bipartite graph $B = (O, N(O), E_B)$ with edge set $E_B = \{uv \mid u \in O \wedge v \in N(O)\}$.

Let I collect all vertices in O that are not matched by M_2 .

while $I' \neq I$ **do**

$I' \leftarrow I; H \leftarrow N(I)$.

$I \leftarrow I \cup \{u \in O \mid \exists v \in H (uv \in M_2)\}$.

end while

else

$I \leftarrow \emptyset$.

end if

Algorithm 2 Using crowns for kernels in the case of VC

Input(s): an instance (G, k) of VC

Output(s): \times if the VC instance has no solution OR a crown $I \subseteq V(G)$ such that

$$|V(G - N[I])| \leq 3k$$

Greedily find a maximal matching M_1 of G .

if $|M_1| > k$ **then**

 return \times

else

 Let O (outsiders) be the set of vertices from G that are not in M_1 .

 ...

 return I

end if

Notation: Größe einer kleinstmöglichen Knotenüberdeckung von G : $vc(G)$

Zur Korrektheit der Kronenregel

Ist I Krone in $G = (V, E)$ mit Kopf $H = N(I)$, so gilt:

$$vc(G) = vc(G - N[I]) + |H|.$$

Da I Krone, ist H kleinstmögliche Knotenüberdeckung von $G[N[I]]$.

Ist C^* kleinstmögliche Knotenüberdeckung (MVC) von G , so ist $C^* \cap N[I]$ VC von $G[N[I]]$ und $C^* \setminus N[I]$ ist VC von $G - N[I]$.

$$\leadsto vc(G) = |C^*| = |C^* \setminus N[I]| + |C^* \cap N[I]| \geq vc(G') + |H|.$$

Ist C' MVC on G' , setze $\tilde{C} = C' \cup H$. \tilde{C} ist VC von G .

$$\leadsto vc(G) \leq |\tilde{C}| = vc(G') + |H|.$$

Bem.: "O.E." \tilde{C} statt C^* , also o.E.: $H \subseteq C^*$.

Bemerkungen zum Kronenalgorithmus (Alternative ?!), falls keine \times -Instanz:

O.E.: Graph G hat keine isolierten Knoten.

Maximales Matching M_1 erfüllt $|M_1| \leq k$.

$O = V \setminus V(M_1)$ ist unabhängige Menge.

Gilt $|O| \leq k$, so haben wir Kern: $|V| = 2|M_1| + |O| \leq 3k$.

Maximum Matching M_2 in bipartitem Graph $B = (O, N(O), E_B)$ erfüllt $|M_2| \leq k$.

Gilt $|M_2| = |N(O)|$, so ist O Krone in G .

\leadsto "Neustart" nach Anwendung der Kronenregel.

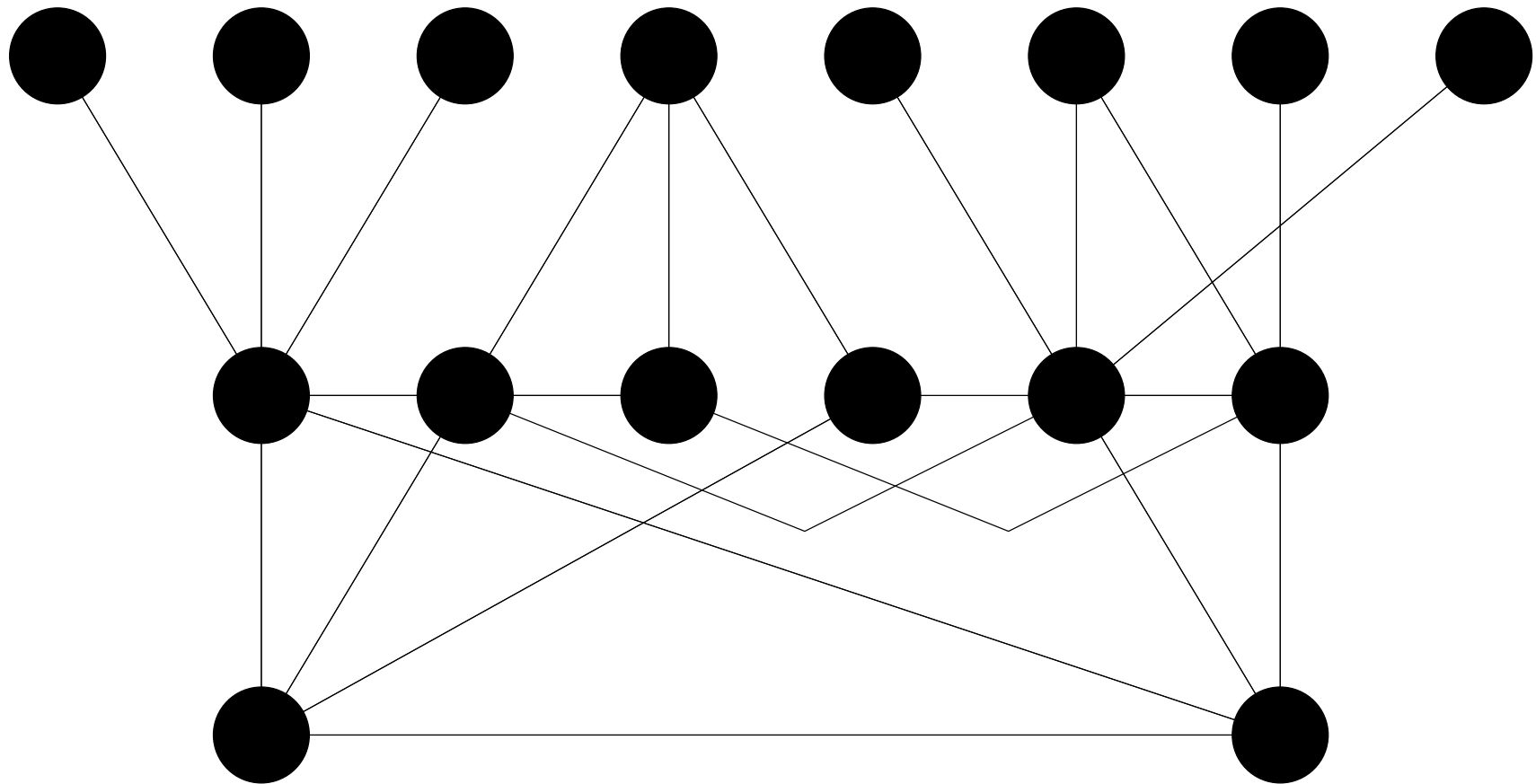
Im übrigen Fall: $|M_2| \leq k$ und $|M_2| < |N(O)|$.

$I = O \setminus V(M_2)$ ist unabhängige Menge, nicht-leer wegen $|O| > k$.

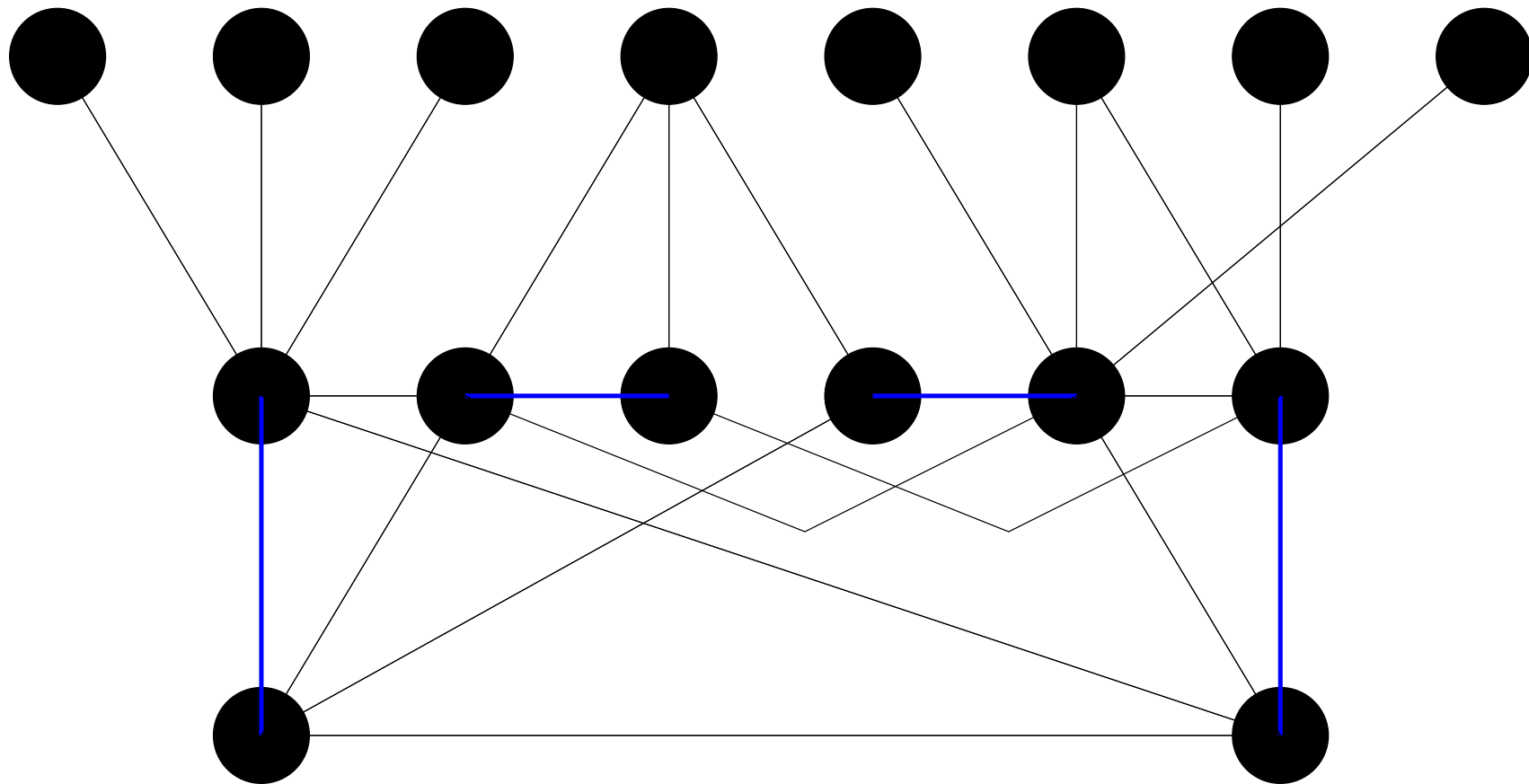
Wähle $x \in I$ beliebig und konstruiere von x startenden M_2 -alternierenden Pfad P ;
 P induziert in B (somit in G) eine Krone. (Hier steckt die Alternative.)

\leadsto "Neustart" nach Anwendung der Kronenregel.

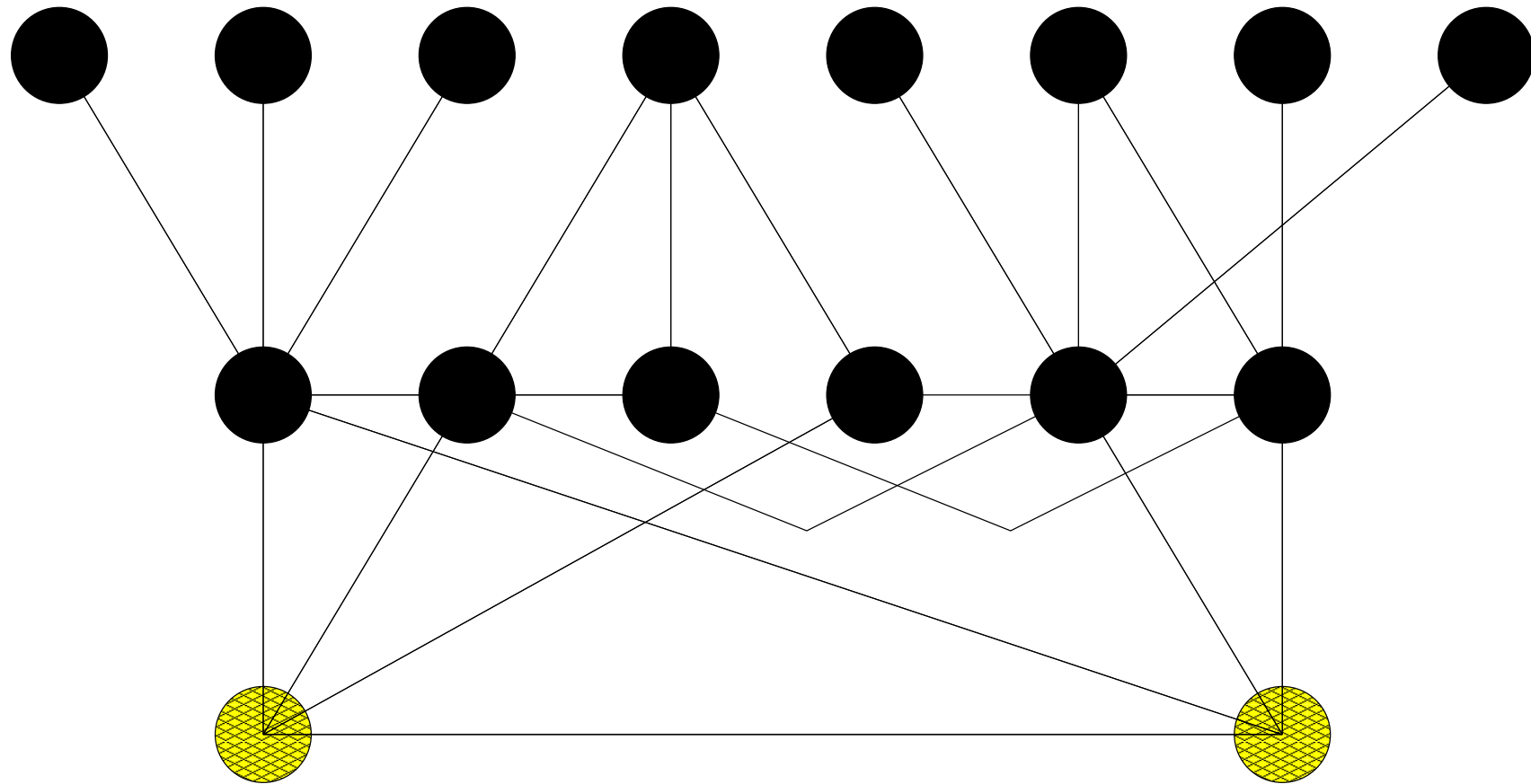
Die Kronenregel an einem Beispiel



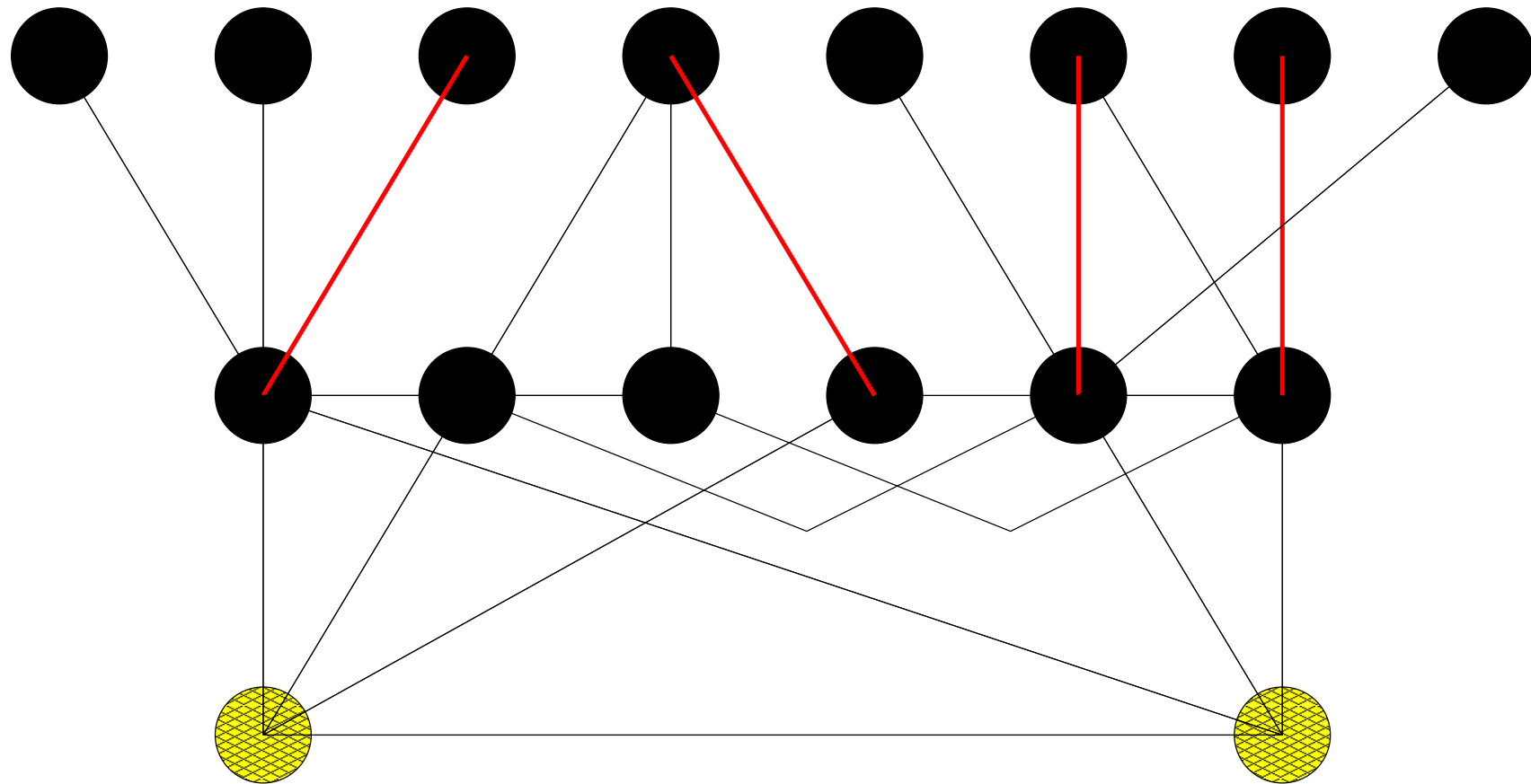
Die Kronenregel an einem Beispiel: Ein maximales Matching M_1



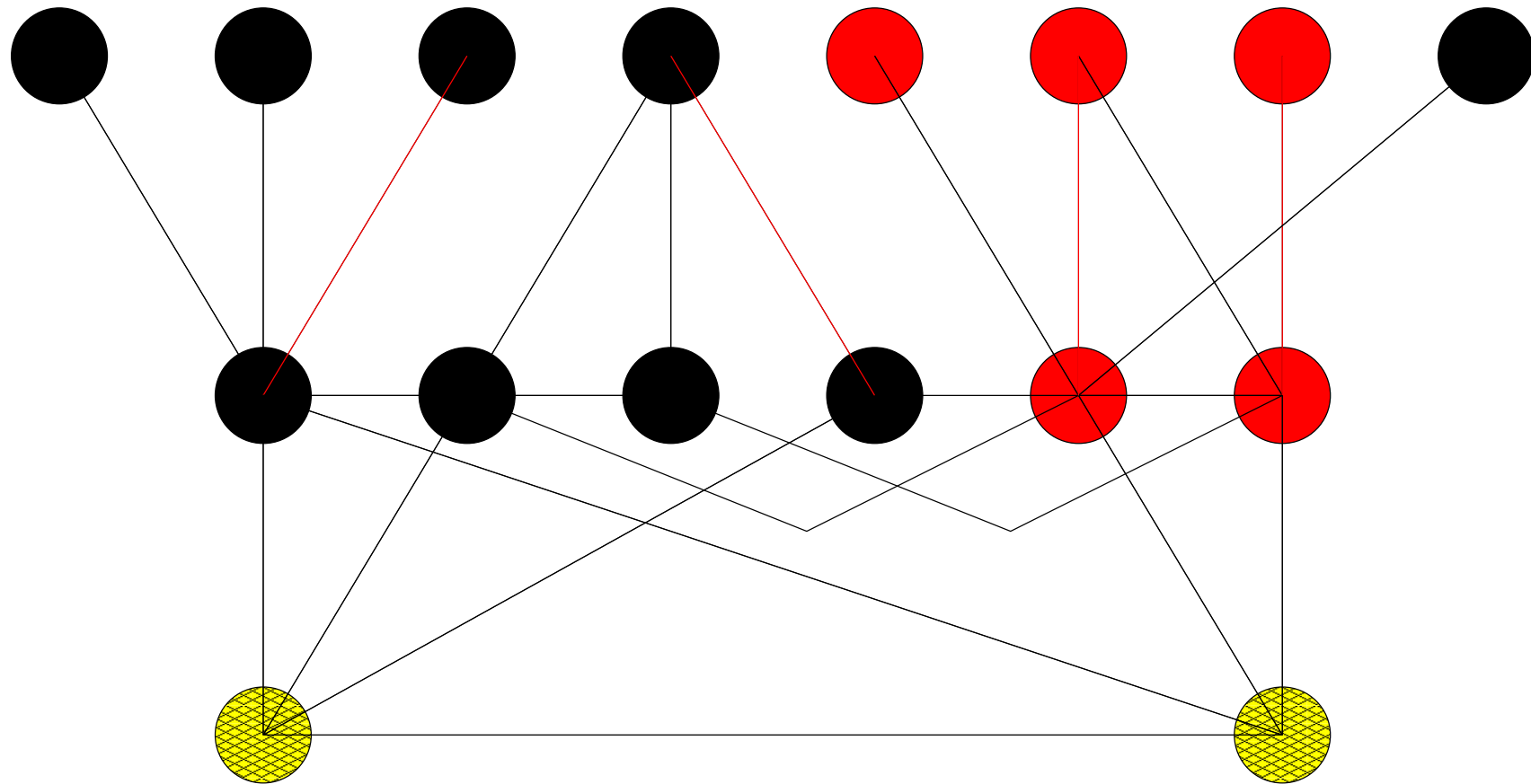
Die Kronenregel an einem Beispiel: $O \cup N(O)$ ist nicht gelb



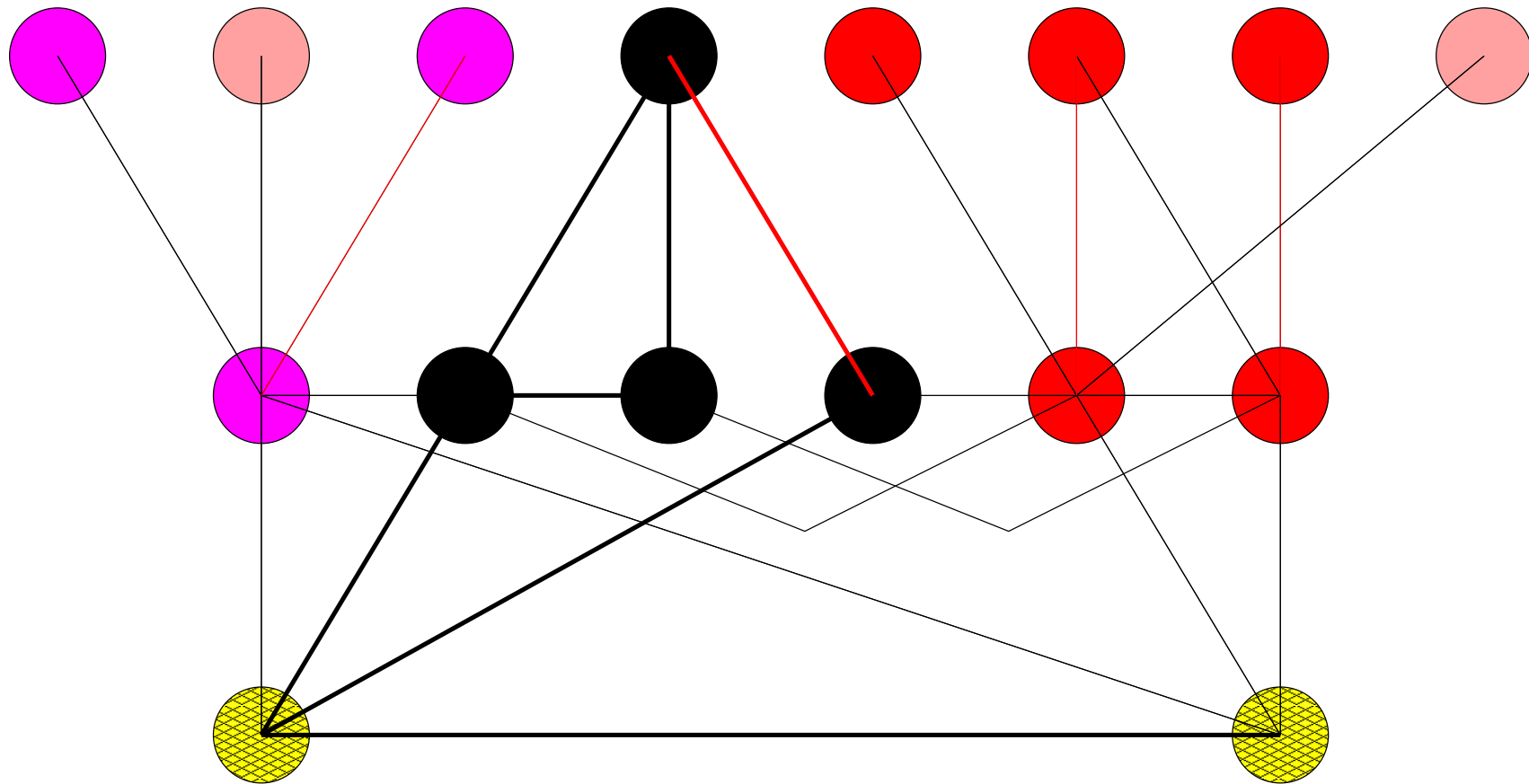
Die Kronenregel an einem Beispiel: Ein rotes Maximum Matching



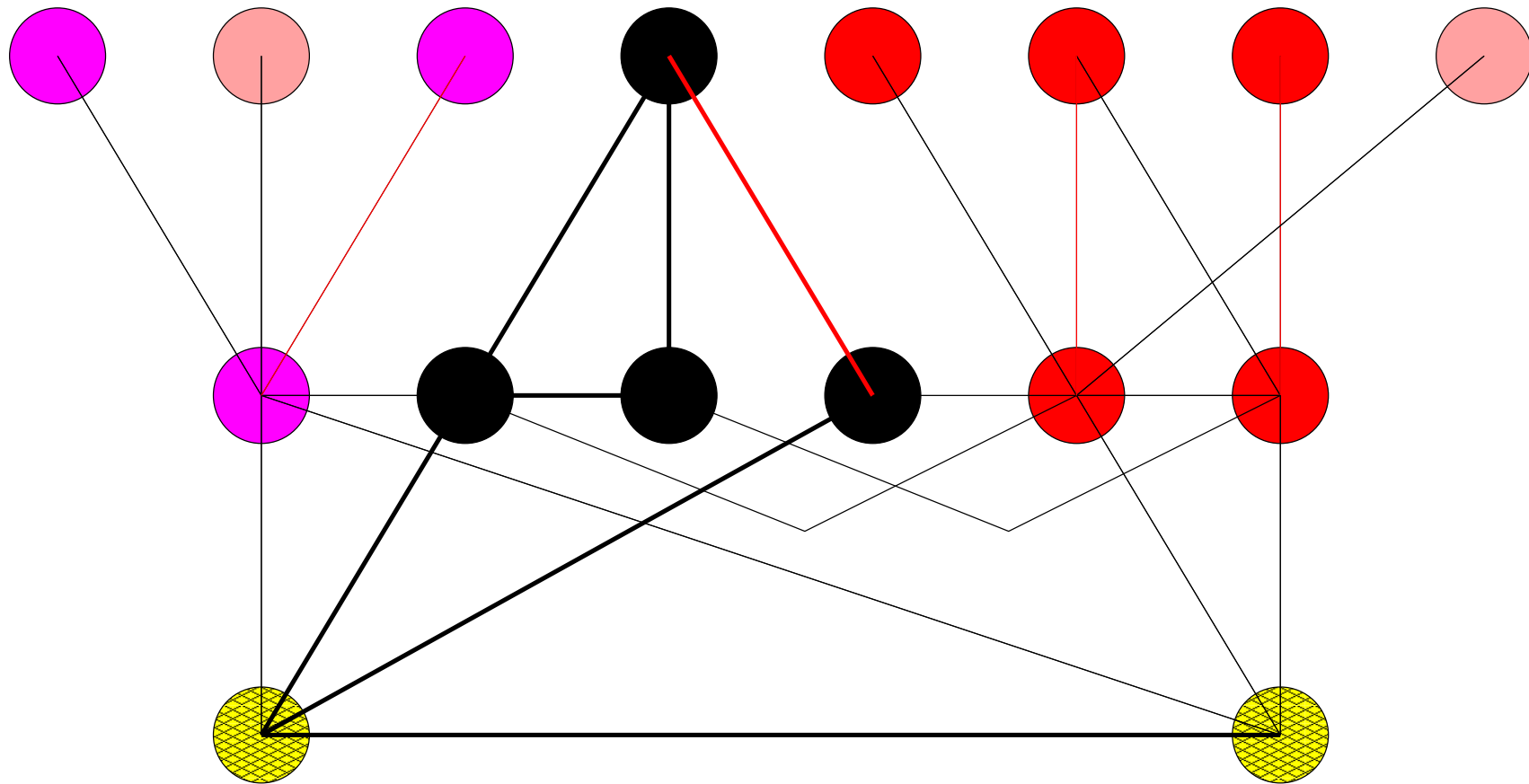
Die Kronenregel an einem Beispiel: Die roten Knoten sind Krone und Kopf



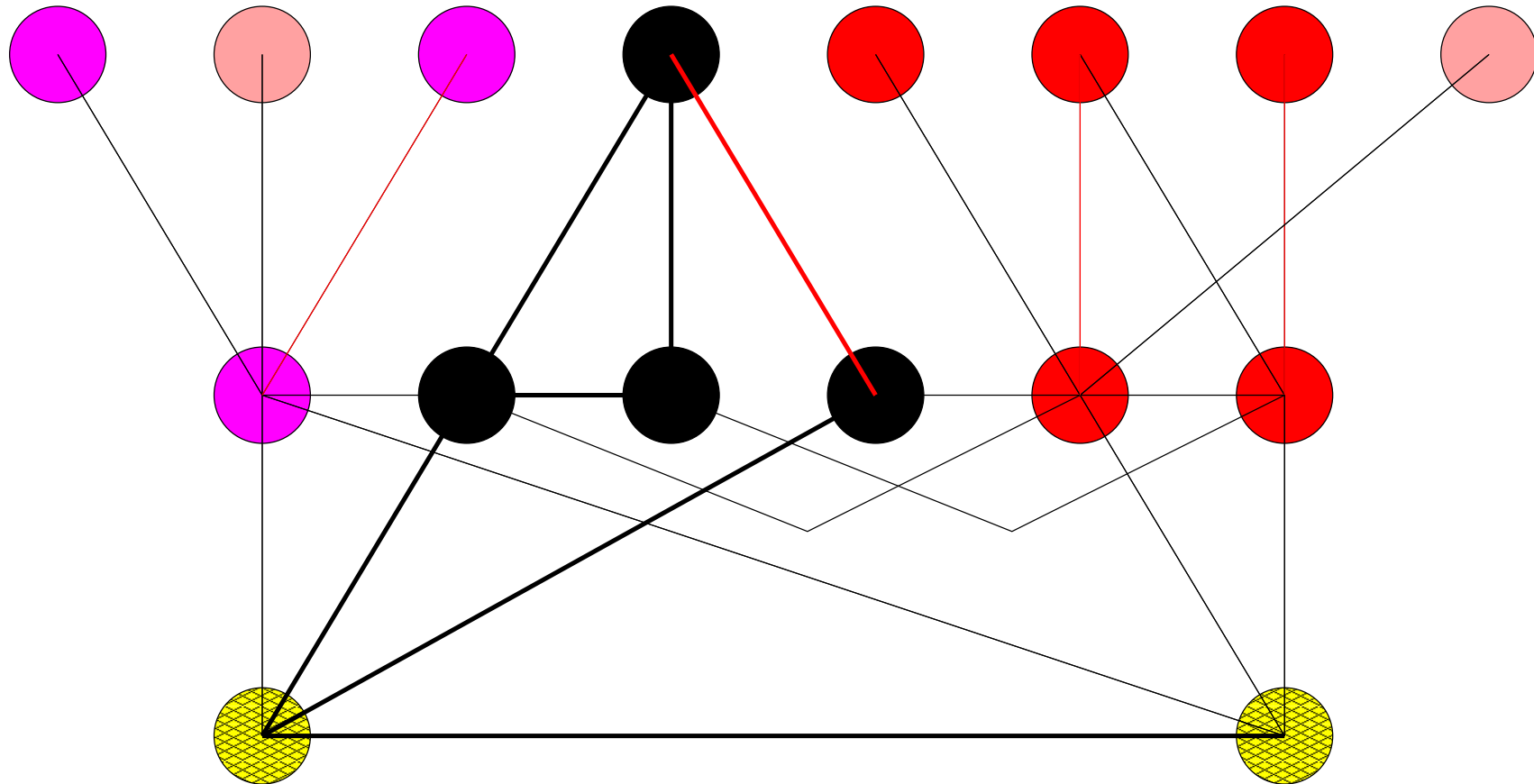
Die Kronenregel am Beispiel: Magenta Knoten bilden noch eine Krone mit Kopf



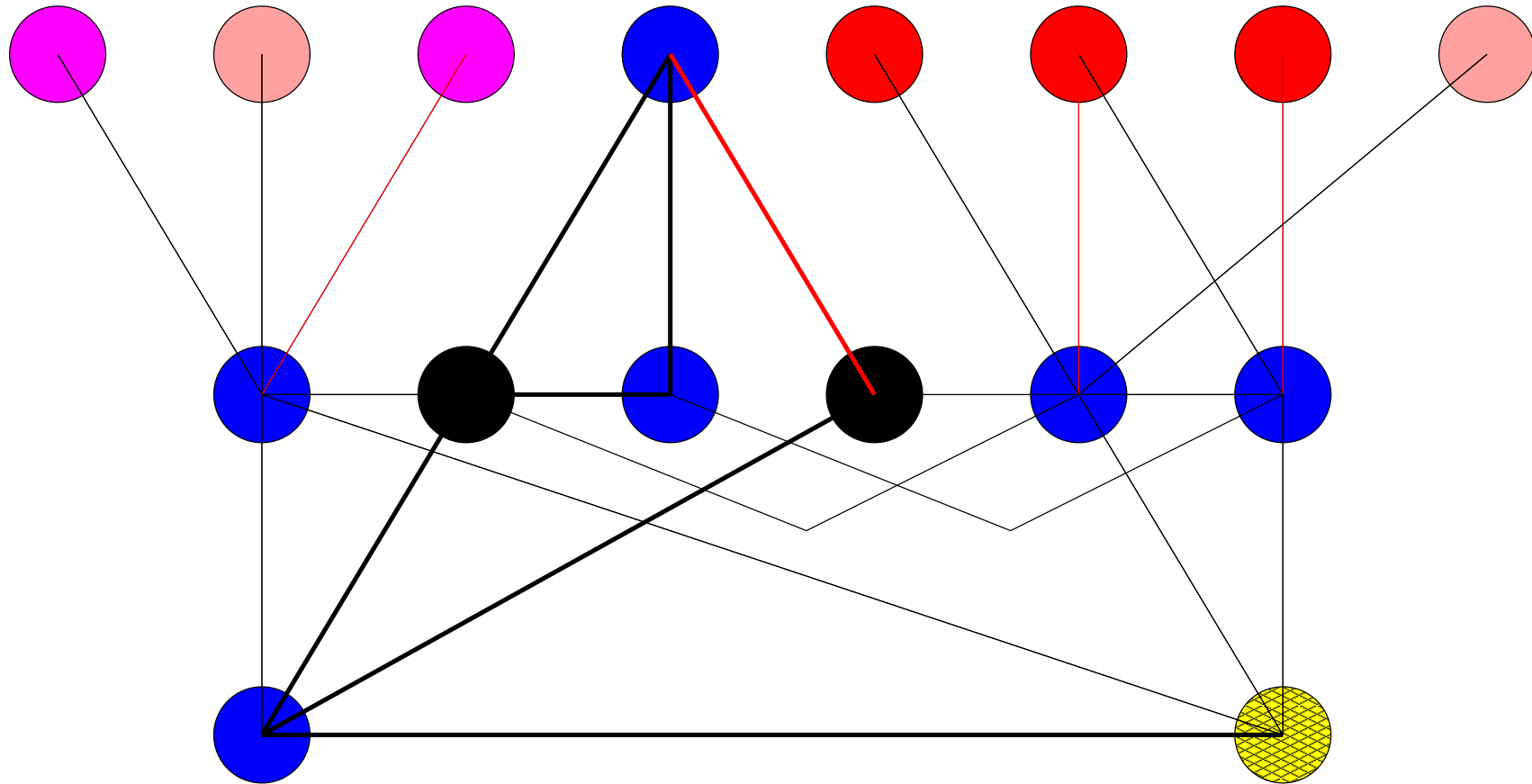
Die Kronenregel an einem Beispiel: Die rosa Knoten werden entfernt (isoliert)



Kern "zunächst": fette Kanten; Achtung: Regeln werden noch dreimal ausgelöst



Die blauen Knoten bilden kleinstmögliche Knotenüberdeckung (Kronenregeln)



Abschließende Diskussion: Welcher Kern ist der beste (für VC)?

- Kriterium Kerngröße: Nemhauser-Trotter
- Von der Laufzeit: Buss-Regel, dann Kronenregel
- Denn: Aufsuchen maximaler Matchings in Linearzeit, (bipartite) Maximum Matchings viel “teurer”
- daher praktischerweise:
Wende zunächst Buss-Regel an, kombiniert mit Regeln für isolierte Knoten und Blätter;
Dann Suche nach Kronen (evtl. entstehen so auch wieder großgradige Knoten ...)
Schließlich NT-Regel (auch hier “Restart” möglich)
- **Faustregel:**
Solange das Problem in Polynomzeit verkleinert werden kann, tue es!

Anwendung für Kronen

- Kronen scheinen sehr speziell zu sein (für das VC-Problem).
- Tatsächlich kann man diese Regeln in vielen “ähnlichen Situationen” anpassen. Zwei konkrete Beispiele folgen:

- P_3 -PACKUNG, oder auch: P_3 -KNOTENÜBERDECKUNG

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

Parameter: natürliche Zahl k

Frage: Gibt es eine Menge von höchstens k Knoten, deren Entfernung keinen Weg der Länge zwei in G belässt?

Etwas fortgeschrittene Kombinatorik:

Das Sonnenblumenlemma von Erdős und Rado



Es sei M eine Menge und $\mathcal{M} \subseteq 2^M$ ein Mengensystem.

Def.: Ein Teilsystem $\mathcal{S} \subseteq \mathcal{M}$ mit $\emptyset \notin \mathcal{S}$ heißt **Sonnenblume** mit **Kern** K gdw.
 $\forall X, Y \in \mathcal{S} : X \cap Y = K \vee X = Y$. Für $X \in \mathcal{S}$ heißt $X \setminus K$ auch **Blütenblatt**.

Satz 2 *Gibt es ein $s \geq 1$, sodass $\forall X \in \mathcal{M} : |X| = s$, und gilt $|\mathcal{M}| > s!(k-1)^s$, so enthält \mathcal{M} eine Sonnenblume $\mathcal{S} \subseteq \mathcal{M}$ mit wenigstens k Blütenblättern.*

Anwendung: Betrachte HITTING SET (mit Parameter k) auf einem Hypergraph mit lauter 3-elementigen Kanten. Enthält der Hypergraph mehr als $6 \cdot k^3$ viele Kanten, so gibt es eine Sonnenblume mit mehr als k Blütenblättern.

Solch eine Sonnenblume kann man durch ihren Kern ersetzen (entspricht der Buss Regel bei VC). Wie lautet also die Reduktionsregel und wie(so) liefert das einen Kern (welcher Größe)?

Beweis: Wir führen einen Induktionsbeweis über s .

Für $s = 1$ (IA) enthält \mathcal{M} nur einelementige Mengen. Gilt $|\mathcal{M}| > (k - 1)$, so ist \mathcal{M} selbst eine Sonnenblume mit leerem Kern und wenigstens k Blütenblättern. Es sei die Behauptung für $s < t$ gezeigt (IV). Wir zeigen im IS die Gültigkeit für $s = t$.

Betrachte ein Teilsystem $\mathcal{T} \subseteq \mathcal{M}$ mit:

- $\forall X, Y \in \mathcal{T} : X \cap Y = \emptyset \vee X = Y$ sowie
- $\forall Z \in \mathcal{M} \setminus \mathcal{T} \exists X \in \mathcal{T} : Z \cap X \neq \emptyset$.

Gilt $|\mathcal{T}| \geq k$, so ist \mathcal{T} eine Sonnenblume mit leerem Kern und wenigstens k Blütenblättern.

Betrachten wir jetzt den Fall $|\mathcal{T}| < k$.

Bilde die Menge $B = \cup\{A \mid A \in \mathcal{T}\}$. Nach Konstruktion gilt: $|B| \leq t \cdot (k - 1)$.

Nach dem Schubfachprinzip gibt es ein Element $x \in B$, das in wenigstens

$$\frac{|\mathcal{M}|}{|B|} > \frac{t!(k-1)^t}{t(k-1)} = (t-1)!(k-1)^{(t-1)}$$

vielen Mengen aus \mathcal{M} enthalten ist. Betrachte das Mengensystem

$$\mathcal{M}_x := \{A \setminus \{x\} \mid A \in \mathcal{M} \wedge x \in A\}.$$

Nach Konstruktion gilt $|\mathcal{M}_x| > (t-1)!(k-1)^{(t-1)}$ und $\forall A \in \mathcal{M}_x : |A| = t-1$.

Wir können hierauf also IV anwenden:

\mathcal{M}_x enthält eine Sonnenblume \mathcal{S}_x mit Kern K_x und wenigstens k Blütenblättern.

Damit ist $\mathcal{S} = \{A \cup \{x\} \mid A \in \mathcal{S}_x\}$ eine Sonnenblume in \mathcal{M} mit Kern $K = K_x \cup \{x\}$ mit wenigstens k Blütenblättern. \square

Kleine Kerne I

Sei $\text{size}(I)$ ein Instanzgrößenmaß für ein parameterisiertes Problem (P, κ) mit:

(a) Probleme mit $\text{size}(I) = \mathcal{O}(1)$ lassen sich in konstanter Zeit lösen.

(b) $\text{size}(I) \geq \kappa(I)$.

Solche Instanzgrößenmaße nennen wir **zulässig**.

Satz 3 *Es sei (P, κ) ein parameterisiertes Problem und s ein zulässiges Instanzgrößenmaß. Gibt es eine Kernreduktion, die zu (I, k) eine Instanz (I', k') liefert mit $\text{size}(I') \leq \alpha k$ für eine Konstante $\alpha < 1$, so liegt (P, κ) in P .*

Idee: Nach $q = \lceil \log_{\alpha}(k) \rceil$ Aufrufen der (Polynomzeit-)Kernreduktion gilt $\text{size}(I') = \mathcal{O}(1)$ für die schließlich erhaltene Instanz (I', k') , denn per Induktion ist $\kappa(I') = k' \leq \alpha^q k$. \square

Beispiel: Das übliche Instanzgrößenmaß für das Knotenüberdeckungsproblem ist die Knotenanzahl; dieses Maß ist zulässig. Es gibt also keinen Problemkern der Größe $0,9999 \cdot k$ für VERTEX COVER, falls $NP \neq P$.

Kleine Kerne II

Ist $\text{size}()$ ein zulässiges Instanzgrößenmaß für das Problem (P, κ) , so heißt $k_d = \text{size}(I) - k$ der **duale Parameter** und demgemäß (P, κ_d) das **duale Problem**.

Beispiel: VERTEX COVER und INDEPENDENT SET sind dual bzgl. des Knotenanzahlmaßes.

Satz 4 *Es seien (P, κ) und (P, κ_d) duale Probleme bzgl. des zulässigen Maßes $\text{size}()$ mit zugehöriger NP-harter klassischer Sprache. Gibt es eine Kernreduktion r , die zu (I, k) eine Instanz (I', k') liefert mit $\text{size}(I') \leq \alpha k$ und eine Kernreduktion r_d , die zu (I, k_d) eine Instanz (I'_d, k'_d) liefert mit $\text{size}(I'_d) \leq \alpha_d k_d$, so gilt $(\alpha - 1)(\alpha_d - 1) \geq 1$ oder $P = \text{NP}$.*

Betrachte nämlich folgenden Algorithmus; er liefert Instanz \tilde{I} mit $\text{size}(\tilde{I}) < \text{size}(I)$, falls $(\alpha - 1)(\alpha_d - 1) = \alpha \cdot \alpha_d - \alpha - \alpha_d + 1 < 1$, also $\alpha \cdot \alpha_d < \alpha + \alpha_d$.
if $k \leq \frac{\alpha_d}{\alpha + \alpha_d} \text{size}(I)$ **then** compute $\tilde{I} \leftarrow r(I, k)$; **else** compute $\tilde{I} \leftarrow r_d(I, \text{size}(I) - k)$.

Beispiel: Für INDEPENDENT SET auf planaren Graphen gibt es keinen Kern kleiner als $2k_d$ unter der Annahme $\text{NP} \neq \text{P}$, denn wir kennen $2k$ -(Knoten)-Kern für VC.

Polynomielle Kerne

Seit 2008 gibt es etliche Arbeiten, in denen gezeigt wird, dass unter gewissen Annahmen (meist schwächer als $P \neq NP$, typischerweise über den Kollaps der Polynomialzeithierarchie auf die dritte Stufe) nachweisen, dass für bestimmte Probleme KEINE polynomiell kleinen Kerne existieren können, die durch eine echte Kernreduktion entstehen.

Beispiele (ohne Beweis):

CVC (Connected Vertex Cover): Knotenüberdeckung ist zusammenhängend

TVC (Total Vertex Cover): Knotenüberdeckung induziert Graph ohne isolierte Knoten

und viele mehr.

Zur Übung: Wieso gibt es Kerne der Größe c^k für CVC oder TVC?

Ein weiteres Beispiel

EDGE CLIQUE COVER ECC

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

Parameter: natürliche Zahl k

Frage: Gibt es eine Menge von höchstens k Cliques C_1, \dots, C_k in G , sodass jede Kante von G in wenigstens einer Clique enthalten ist?

Durch erschöpfendes Löschen von Knoten x , falls es einen weiteren Knoten y gibt mit $N[x] = N[y]$, erhält man einen Graphen, bei dem NEIN geantwortet werden kann, falls er mehr als 2^k Knoten enthält. Wir können nämlich zu jedem Knoten z einen k -dimensionalen Bitvektor $b^z = (b_1^z, \dots, b_k^z)$ assoziieren, wobei $b_i^z = 1$, falls z in der Clique C_i enthalten ist (Knoten können zu mehreren Cliques gehören). Gäbe es zwei Knoten y, z mit $b^y = b^z$, so wäre $N[z] = N[y]$, was bei reduzierten Graphen nicht vorkommt.

Falls es einen $\mathcal{O}^*(2^{o(k)})$ -Kern oder einen $\mathcal{O}^*(2^{2^{o(k)}})$ -Algorithmus für ECC gibt, existiert auch eine Möglichkeit, SAT schneller als $2^{o(n)}$ zu lösen für Instanzen mit n Variablen. Das wird gemeinhin nicht geglaubt (**Exponentialzeithypothese**, ETH). Dies wurde in Arbeiten auf der ICALP 2012 und SODA 2013 gezeigt.

Sind Kerne immer polynomiell klein? Ein konkreter Fall

Betrachten wir im Folgenden ein “Pfad-Problem”:

ROOTED LONGEST PATH RLP

Eingabe: gerichteter Graph $G = (V, E)$, Wurzelknoten r

Parameter: natürliche Zahl k

Frage: Gibt es einen (gerichteten) Pfad der Länge k in G , der bei r startet?

Das Problem ist NP-vollständig, kann aber in Zeit $\mathcal{O}^*(c^k)$ für eine Zahl c gelöst werden (Farbkodieren), und hat also einen Kern der Größe c^k .

Gibt es einen Kern polynomieller Größe für dieses Problem?

Satz 5 Falls $P \neq NP$, hat RLP keinen Kern polynomieller Größe, der durch eine echte Kernreduktion entsteht.

Beweis: Angenommen, für RLP gibt es eine echte Kernreduktion ρ , die einen Kern polynomieller Größe erzeugt, also $\rho(G, r, k) = (G', r', k')$ mit $\text{size}(G') \leq p(k)$ für ein Polynom p und $\rho(k) = k' \leq k$.

Gibt es eine Kante $r\vec{u}$ in G , so entsteht $G[r \leftarrow ru]$ aus G durch

- (a) Löschen von r und aller inzidenten Kanten,
- (b) Umbenennen von u in r und
- (c) Umbenennen von Knoten x in $\langle x, u \rangle$ für alle $x \neq r$.

Es gilt: (G, r, k) ist ✓-Instanz von RLP gdw.

für ein $u \in N^+(r)$ gilt: $(G[r \leftarrow ru], r, k - 1)$ ist ✓-Instanz von RLP gdw.

$(\bigcup_{u \in N^+(r)} G[r \leftarrow ru], r, k - 1)$ ist ✓-Instanz von RLP (Achtung: "Verleimen" bei r) gdw.

$(\rho(\bigcup_{u \in N^+(r)} G[r \leftarrow ru]), \rho(r), \rho(k - 1))$ ist ✓-Instanz von RLP.

Da ρ echte Kernreduktion ist, gilt: Iterieren wir dies (höchstens) $k - 1$ -mal, so erhalten wir eine triviale Frage für einen "polynomiell kleinen" Graphen.

Das würde einen deterministischen Polynomzeitalgorithmus für RLP ergeben.

Also wäre $P = NP$. $\nabla \square$

Noch ein Übungsbeispiel für große Kerne

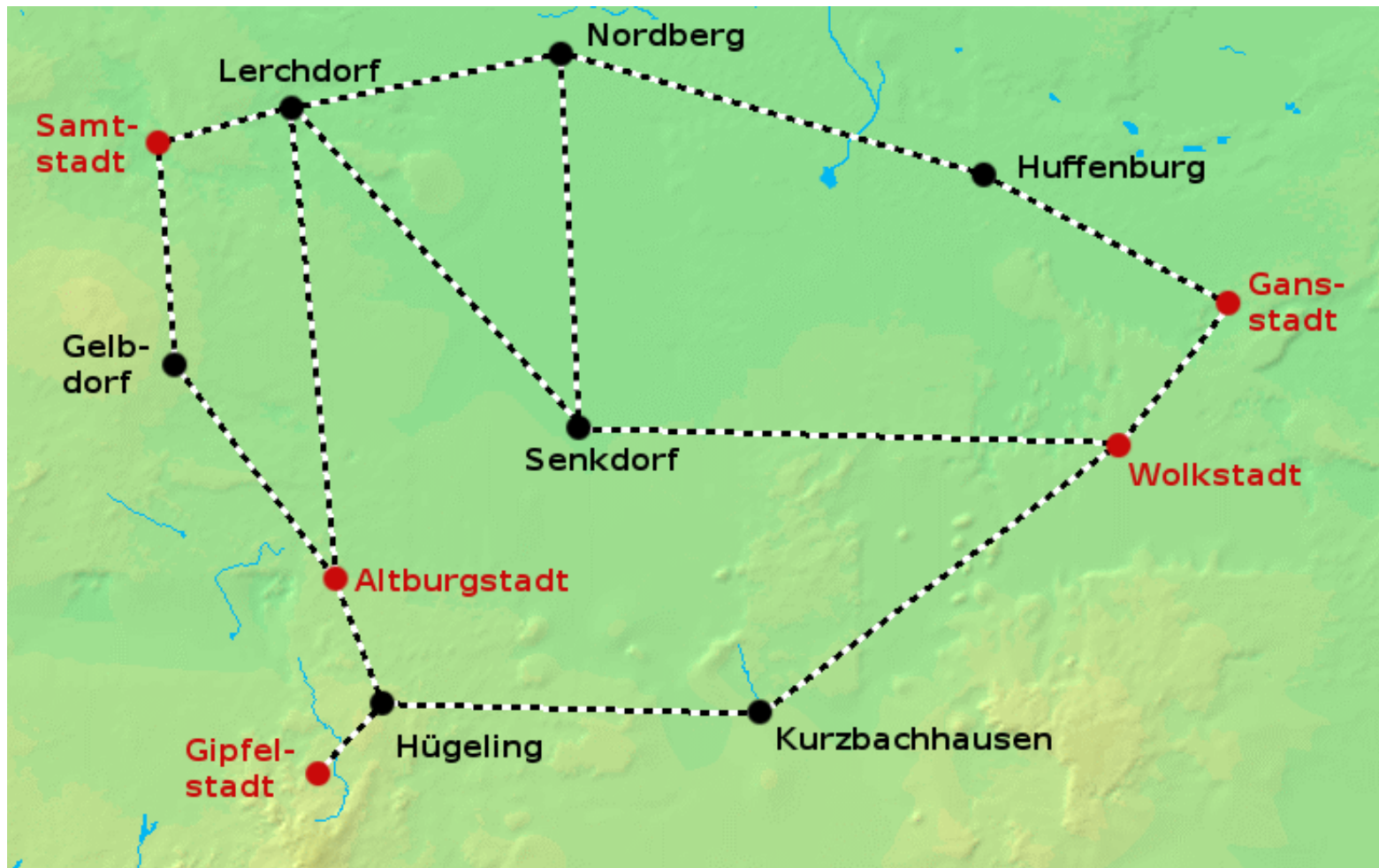
Es sei $G = (V, E)$ ein ungerichteter Graph.

- Erinnerung / Hilfsdefinitionen: $N[v] := N(v) \cup \{v\}$ bezeichnet die *abgeschlossene Nachbarschaft* des Knoten $v \in V$. Allgemeiner sei $N[U] := \bigcup_{v \in U} N[v]$ die abgeschlossene Nachbarschaft der Knotenmenge $U \subseteq V$.
- Eine nicht-leere Knotenmenge S heißt *defensive Allianz* genau dann, wenn jeder Knoten v in S nicht mehr Nachbarn besitzt, die nicht in S liegen, als solche, die in S liegen; formal bedeutet dies:

$$\forall v \in S : |N[v] \cap S| \geq |N[v] \setminus S|.$$

Intuitiv modelliert dieser Begriff, dass die Allianz S gewährleistet, dass in keinem Fall die Zahl der potentiellen benachbarten Feinde diejenige der benachbarten Freunde übertrifft und somit immer eine gute Verteidigungsmöglichkeit besteht.

- Ist eine Allianz S auch noch eine dominierende Menge in (V, E) , gilt also $N[S] = V$, so heißt diese Allianz *global*.



1. Finden Sie Knotenmengen S_1, S_2, S_3 in dem in der Abb. angegebenen Graphen mit:
 - S_1 ist eine defensive Allianz mit wenigstens zwei Knoten, die nicht global ist .
 - S_2 ist eine dominierende Menge, die keine defensive Allianz bildet.
 - S_3 ist eine kleinstmögliche globale defensive Allianz.
2. Aus den soeben eingeführten Allianz-Begriffen ergeben sich zwei Minimierungsprobleme, beispielsweise die Aufgabe, in einem vorgegebenen Graphen G eine kleinstmögliche defensive Allianz zu finden. (Beachte: Allianzen sind stets nicht-leere Mengen!) Formalisieren Sie die zugehörigen Entscheidungsprobleme!
3. Sie haben bei der Bearbeitung der vorigen Teilaufgabe zweimal einen “natürlichen Parameter” eingeführt, nämlich als obere Schranke für die gesuchten Allianzen. Suchen Sie Reduktionsregeln für die so parameterisierten Entscheidungsprobleme. Beweisen Sie jeweils die Korrektheit der gefundenen Regeln und machen Sie sich auch Gedanken darüber, ob so ein Reduktionsschritt in polynomieller Zeit abgearbeitet werden kann.
4. Überlegen Sie sich, wie sich aus den von Ihnen gefundenen Reduktionsregeln evtl. Kernreduktionen für die angegebenen Probleme angeben lassen. Wie groß sind die so von Ihnen gefundenen Kerne, gemessen in der Zahl der Knoten der reduzierten Instanz?