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)

  Bsp. Buss-Regel

- Verwendung bekannter (nicht-trivialer) mathematischer Aussagen

  Bsp. Vierfarbensatz; auch NT-Theorem

- Verallgemeinerung bekannter einfacher Regeln

  Kern für 3-HS?

- Dafür häufig wichtig: Betrachtung annotierter Probleme, z.B. Katalysatoren

- Greedy-Algorithmen

Fokus auf Extremen bei Kernreduktionen

Bsp.: Buss' Regel bei VC

MAXIMUM MINIMAL VERTEX COVER MMVC
Eingabe: ungerichteter Graph $G = (V, E)$
Parameter: natürliche Zahl k
Frage: Gibt es eine minimale Knotenüberdeckung der Größe mindestens k?

Wie wäre die Buss-Regel für MMVC umzuformulieren?!

MMVC-Regeln

Reduktionsregel 1 *Lösche isolierte Knoten (Beibehaltung des Parameters)*

Reduktionsregel 2 *Ist v ein Knoten vom Grad größer gleich k, dann JA.*

Satz 1 *MMVC gestattet einen Kern der Größe k^2 (gemessen in der Knotenzahl).*

Wie sieht "vollständiges Regelsystem" aus?

Greedy-Algorithmen für Kernreduktionen

Klappt manchmal bei Maximierungsproblemen

Dann zumeist “Querbezüge” zur Approximation.

Erinnerung planare Graphen

Eulersche Flächenformel / Polyederformel
für zusammenhängende planare Graphen mit wenigstens zwei Knoten

$$f - 2 = m - n$$

n: Knotenzahl
m: Kantenzahl
f: Facettenzahl

Folgerung 2 *In einem planaren Graphen gibt es stets einen Knoten vom Grad höchstens fünf.*

Folgerung 3 *Greedy gestattet einen $6k$ Kern für PIS.*

Was bedeutet das für die Approximation von PIS?

Übungsaufgaben

Frage: MMVC auf planaren Graphen?
MMVC: Gibt es minimale Knotenüberdeckung der Größe mindestens k in G?

Frage: MMIS auf planaren Graphen?
MMIS: Gibt es maximale unabhängige Menge der Größe höchstens k in G?

Algorithm 1 Greedy kernelization for PIS

Input(s): planar graph $G = (V, E)$, positive integer k

Output(s): either independent set I with $|I| = k$ or $|V| \leq 6(k-1)$

```
G′ ← G; i ← 0; I ← ∅
while V(G′) ≠ ∅ and i < k do
    pick vertex v ∈ V(G′) of smallest degree
    G′ ← (G′ − N[v]) {|N[v]| ≤ 6 (∗)}
    i ← i + 1
    I ← I ∪ {v} {I is an independent set of size i in G. [+]}
end while
if i = k then
    I is an independent set of size k in G.
else {V(G′) = ∅ and i < k}
    { (∗) ⇒ ≤ 6(k − 1) vertices have been taken out before exhausting V.}
end if
```

Noch ein Problem. . . (und noch eine Übungsaufgabe)

P_3-PACKUNG
Eingabe: ungerichteter Graph $G = (V, E)$
Parameter: natürliche Zahl k
Frage: Gibt es eine Menge von mindestens k paarweise knotendisjunkten Wegen der Länge zwei in G?

So eine Menge von Wegen heißt auch P_3-Packung.

Welche Objekte bleiben "übrig", wenn man zunächst irgendeine maximale P_3-Packung berechnet? (Greedy!)
Überlegen Sie sich zur **Übung**, ob Ihnen diese (Hilfs-)Struktur zu weiteren Reduktionsregeln inspiriert?

Parameterisierte Dualität

Reparameterisierungsmethode “bekannter” Probleme

Bsp.: VC vs. IS durch Komplementierung.

Bsp.: Nonblocker: Komplement von DS.

$N \subseteq V$ ist Nonblocker-Menge gdw. jeder Knoten in N hat einen Nachbarn außerhalb von N. “Jeder kennt jemanden außerhalb des eigenen Kreises.”

Formaler: NONBLOCKER

Eingabe: ungerichteter Graph $G = (V, E)$ der Ordnung $n = |V|$

Parameter: natürliche Zahl k

Frage: Gibt es eine dominierende Menge der Größe höchstens $n - k$ in G?

Einfache Aussagen zu dominierenden Mengen

Lemma 4 *Ist $G = (V, E)$ ein Graph ohne isolierte Knoten, so ist das Komplement einer minimalen dominierenden Menge eine dominierende Menge.*

Beweis: Sei $D \subseteq V$ eine minimale dominierende Menge von $G = (V, E)$.
Betrachte $x \in D$. Da D minimal und $N(x) \neq \emptyset$, ist "$N(x) \subseteq D$" falsch.
Also gibt es ein $y \in V \setminus D$ mit $y \in N(x)$.
Daher ist $V \setminus D$ eine dominierende Menge. $\square$

Satz 5 (Ore 1962) *Ist $G = (V, E)$ ein Graph ohne isolierte Knoten, so gibt es eine dominierende Menge $D \subseteq V$ mit $|D| \leq |V|/2$.*

Beweis: Es sei D' eine minimale dominierende Menge von $G = (V, E)$.
Mit Lemma 4 ist $V \setminus D'$ ebenfalls eine dominierende Menge.
Sei D die kleinere der Mengen D' und $V \setminus D'$. D erfüllt die Behauptung. $\square$

Wie wird daraus ein Kern für NONBLOCKER?

Folgerung 6 *Es sei (G, k) eine Instanz von* NONBLOCKER*. Wenn Graph G keine isolierten Knoten aber mehr als $2k$ Knoten hat, so ist (G, k) eine ✓-Instanz.*

Frage: Umgang mit isolierten Knoten?

Reduktionsregel 3 *Ist x ein isolierter Knoten, so lösche ihn (und lasse den Parameter unverändert).*

Die Korrektheit dieser Regel folgt mit folgender leichter Aussage:

Lemma 7 *Ist D dominierende Menge von G und x ein isolierter Knoten, so gilt: $x \in D$.*

Satz 8 NONBLOCKER *besitzt einen $2k$ Kern.*

Geht es besser?! (ein typischer “Sport” der Algorithmik)

Versuche, “schlimmste Fälle” zu konstruieren. . .

$\leadsto$ Weitere Reduktionsregeln?
Versuch: Fokus auf Extremen
Weiterer Trick: Einführung von Annotationen

Zurück zu NONBLOCKER: Knoten vom Grad 1 (Blätter)

Erweiterte Instanz (mit Annotation):
Graph mit ausgezeichnetem Knoten c in der dominierenden Menge.
Wegen seiner Funktion heißt c auch Katalysator oder hier speziell Dominator.

(Hinter-)Grund:
Ist x Nachbarknoten eines Blattes, so kann x mit Dominator c zu neuem Dominator $[x, c]$ verschmolzen (identifiziert) und das Blatt entfernt werden.
Die Nachbarschaft von $[x, c]$ ist $N(x) \cup N(c)$.
Diese Regel werden wir in Regel 5 verallgemeinern.

Mehr hilfreiche Mathematik. . . (ohne Beweis)

Satz 9 (Blank 1973; McCuaig / Shepherd 1989) *Hat ein zusammenhängender Graph* $G = (V, E)$ *Minimalgrad zwei und gehört er nicht zu einem von sieben "Ausnahme-Graphen" (jeweils mit maximal sieben Knoten), so besitzt* G *eine dominierende Menge mit höchstens* $2/5 \cdot |V|$ *vielen Knoten.*

Wir können diesen nichttrivialen bekannten Satz ausnutzen für einen Problemkernalgorithmus durch Einführen von Katalysatorregeln und Dekatalysatorregeln.

Die “alten” Regeln (verallgemeinert)

Sei (G, c, k) eine Instanz von NB (annotiert mit Katalysator c).

Reduktionsregel 4 (verallgemeinerte Regel für isolierte Knoten) *Ist* C *eine vollständige Graphkomponente ohne* c*, reduziere zu* $(G - C, c, k - (|C| - 1))$.

Reduktionsregel 5 (verallgemeinerte Blattregel) *Gibt es* $x \in V(G)$ *mit* $U \subseteq N(x)$*, sodass* $N(U) \subseteq U \cup \{x\}$ *und* $c \notin U$*, dann reduziere zu der Instanz* $(G[c = x] - U, [c, x], k - |U|)$.

Zur Übung: Beweisen Sie die Korrektheit der beiden Regeln.

(De-)Katalysatorregeln: Hin zu annotierten Instanzen und zurück

Katalysatorregel Ist (G, k) eine NONBLOCKER-Instanz $G = (V, E)$, dann ist (G', c, k) eine äquivalente annotierte Instanz, wobei $c \notin V$ ein neuer Knoten ist und $G' = (V \cup \{c\}, E)$.

Dekatalysatorregel Sei (G, c, k) eine annotierte NONBLOCKER-Instanz. Die neue nicht-annotierte Instanz (G', k') entsteht wie folgt:

Verbinde c mit drei neuen Knoten u, v und w. Ferner führe neue Kanten uv und vw ein. Alle andere Knoten und Kanten bleiben unverändert. Dies beschreibt den neuen Graphen G'. Setze $k' = k + 3$.

Achtung: Keine echte Kernreduktion !

Algorithm 2 A kernelization algorithm for NB

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

Output(s): an equivalent instance (G', k') of NB with $V(G') \subseteq V(G)$, $|V(G')| \leq 5/3 \cdot k'$ and $k' \leq k + 3$ OR ✓

if G has more than seven vertices **then**
    Apply the catalyzation rule.
    Exhaustively apply Rules 4 and 5 for neighborhoods U up to size two.
    Apply the decatalyzation rule.
    {This leaves us with a reduced instance (G', k').}
    if $|V(G')| > 5/3 \cdot k'$ **then**
        return ✓
    else
        return (G', k')
    end if
else
    Solve by table look-up and answer accordingly.
end if

Dieser Algorithmus zeigt:

Satz 10 NONBLOCKER *besitzt einen Kern mit höchstens* $\frac{5}{3}k + 3$ *Knoten.*

Wir werden im Folgenden Varianten der Idee für den NONBLOCKER-Kern besprechen sowie eine mögliche Anwendung.

Eine alternative Idee für das Einführen des Katalysators

katalytisches Verzweigen statt Katalysatorregel: Eingangs werden n annotierte Instanzen erzeugt durch systematisches Durchprobieren jedes Knotens als Kat.

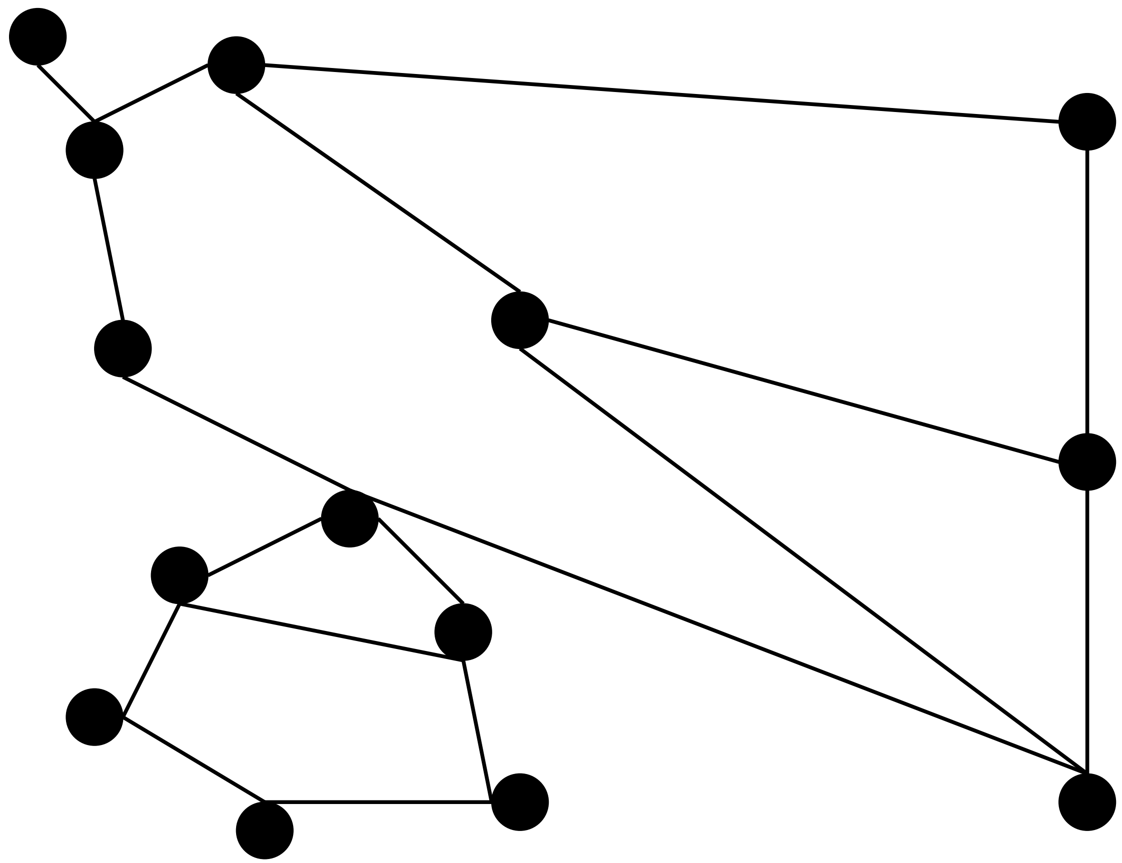
Die folgenden Folien zeigen eine “unglückliche Wahl”. (Katalysator in rot)

Es wird eine weitere Regel benutzt, um sämtliche Ausnahmegraphen abzuhandeln (insgesamt werden damit alle Graphen mit zwei benachbarten Knoten vom Grad zwei reduziert); dies erklärt auch, weshalb die Dekatalysatorregel nicht etwas einfacher gewählt wurde.

Reduktionsregel 6 *Falls in* $G = (V, E)$ $v, v', u, u' \in V$ *vorliegt:*
$\deg(v) = \deg(v') = 2$, $v \neq c$, $v' \neq c$, $N(v) = \{u, v'\}$, $N(v') = \{u', v\}$, $u \neq u'$
$\rightsquigarrow$ *Reduziere zu* $(G[u = u'] - \{v, v'\}, c, k-2)$.

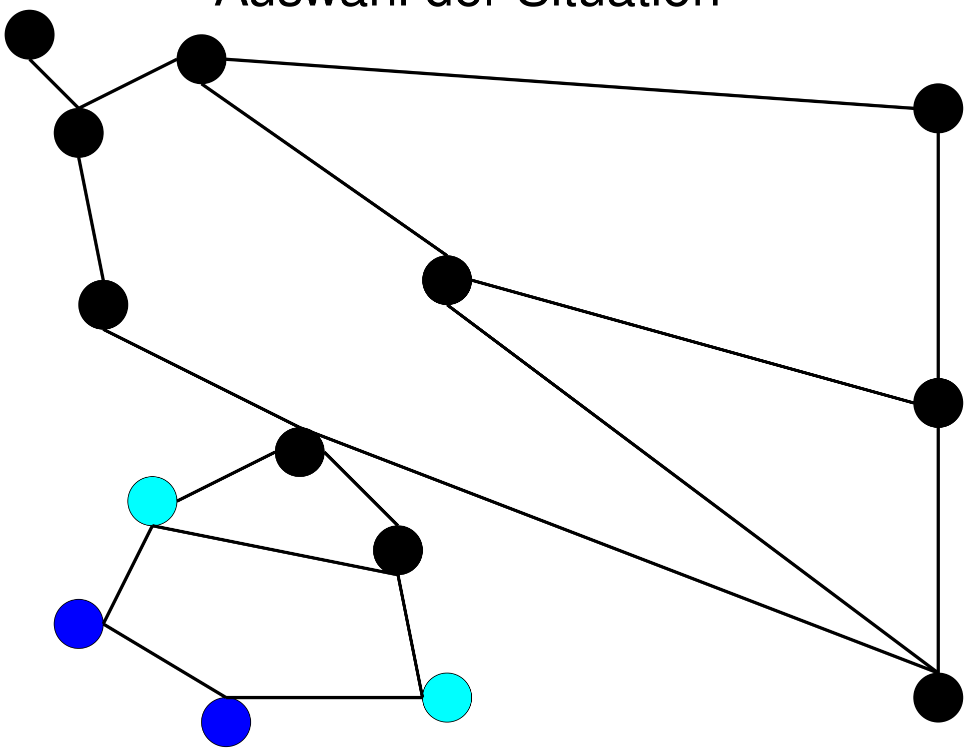
Im Folgenden wird die Arbeit der Reduktionsregeln an einem Beispiel gezeigt; außerdem sieht man, dass z.B. die letzte Regel auch ohne Katalysator arbeitet. Dann wird die Lösung “rückwärts” für den ursprünglichen Graphen erarbeitet und schließlich eine bessere Katalysatorwahl gezeigt.

Der Ausgangsgraph

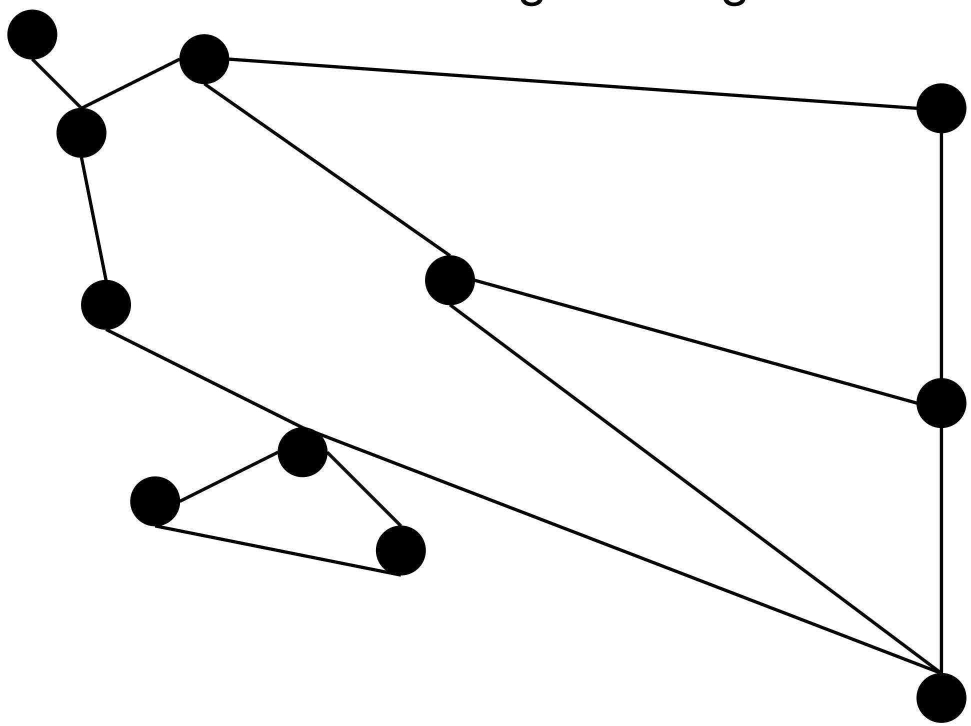


Anwendung von Regel 6

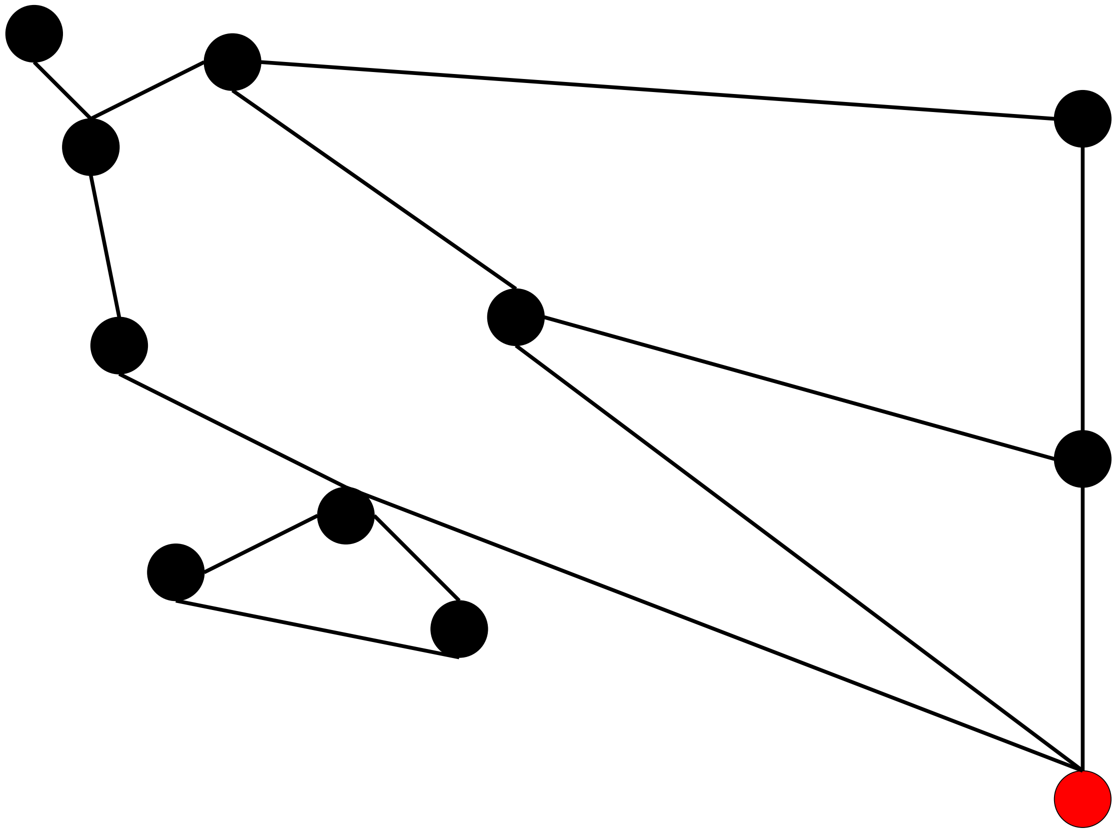
Auswahl der Situation



Nach Anwendung der Regel

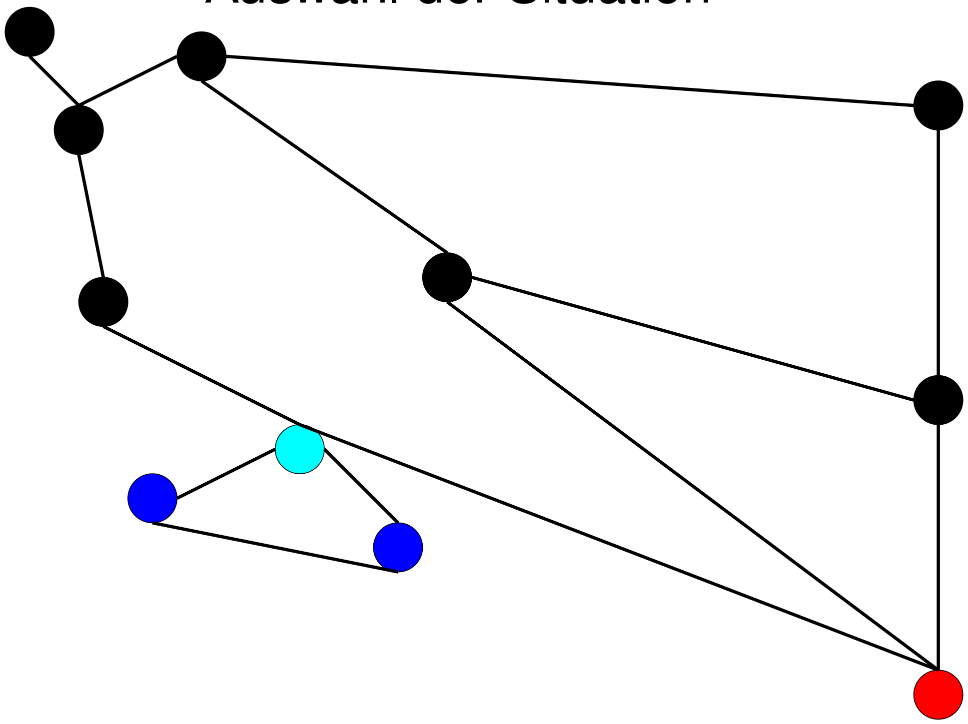


Katalytisches Verzweigen wählt einen Knoten aus

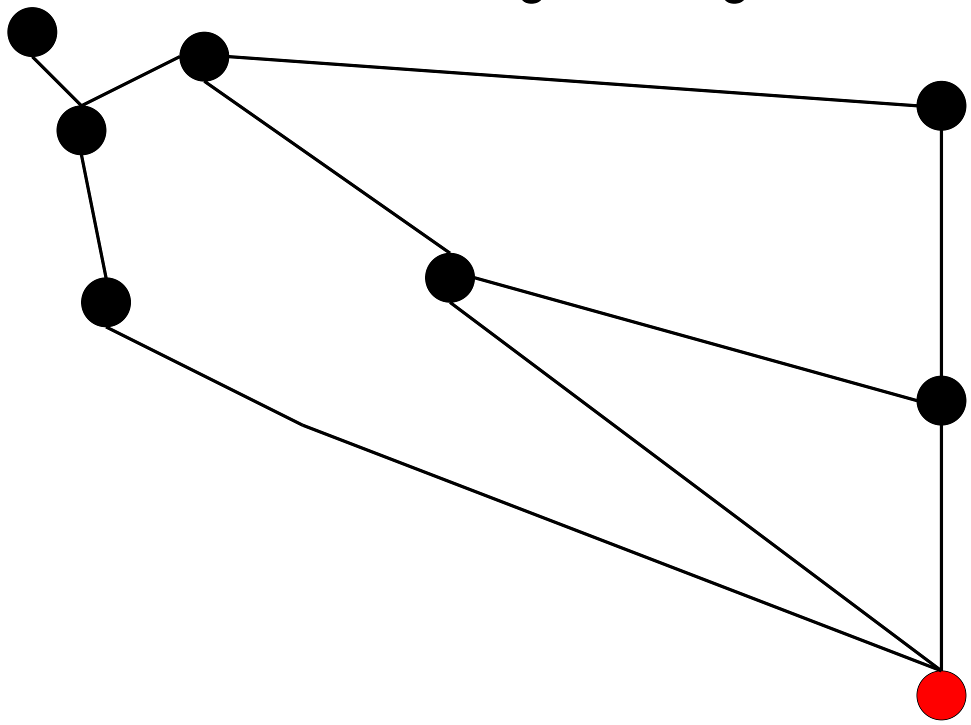


Anwendung der verallgemeinerten Blattregel 5

Auswahl der Situation

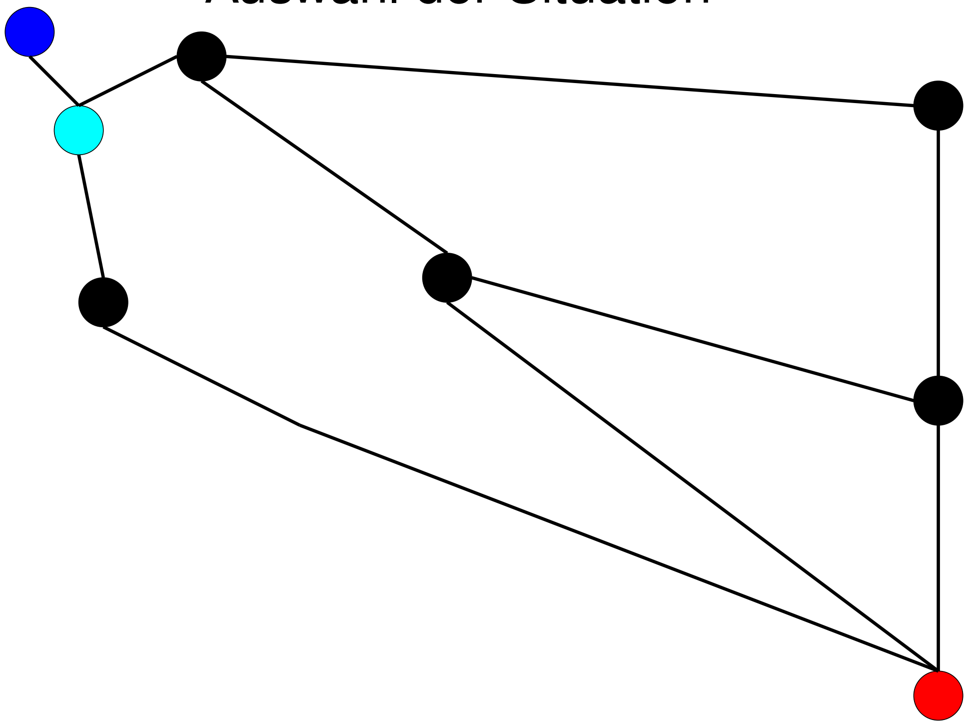


Nach Anwendung der Regel

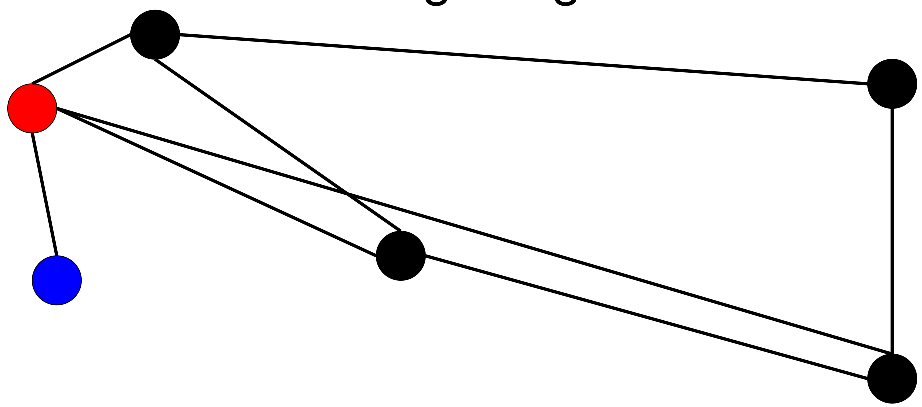


Eine Blattregelanwendung löst die nächste aus

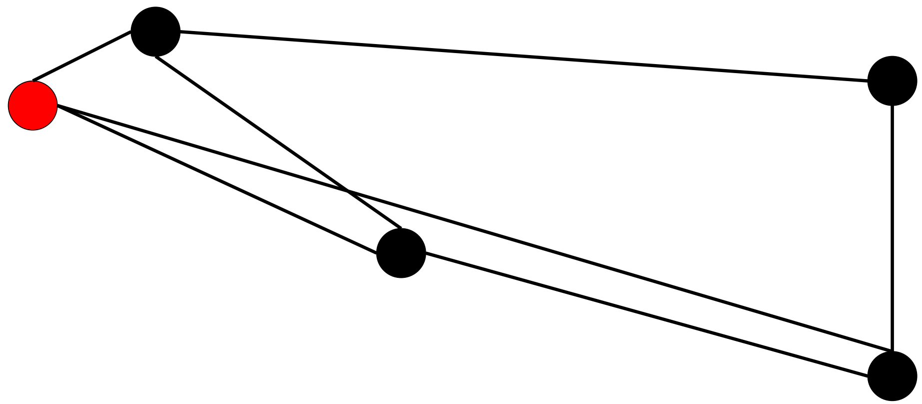
Auswahl der Situation

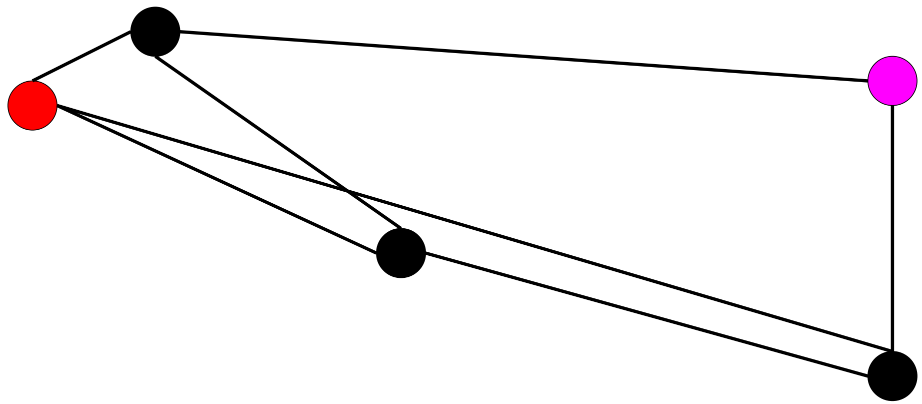


Nach Anwendung der Regel: neue Anwendungsmöglichkeit

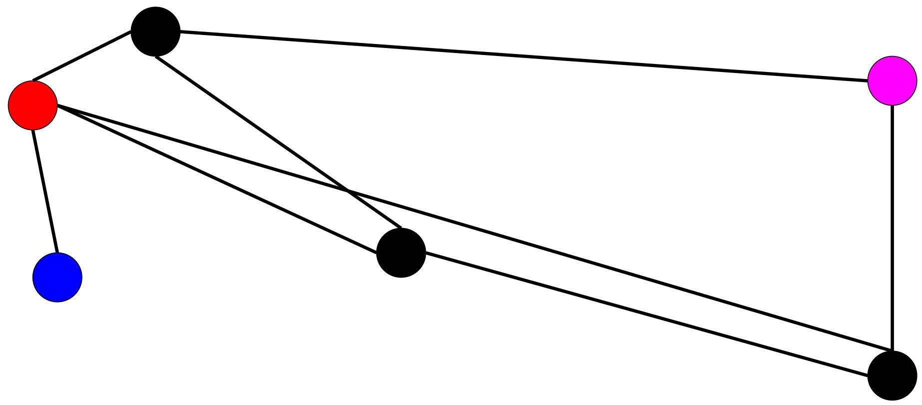


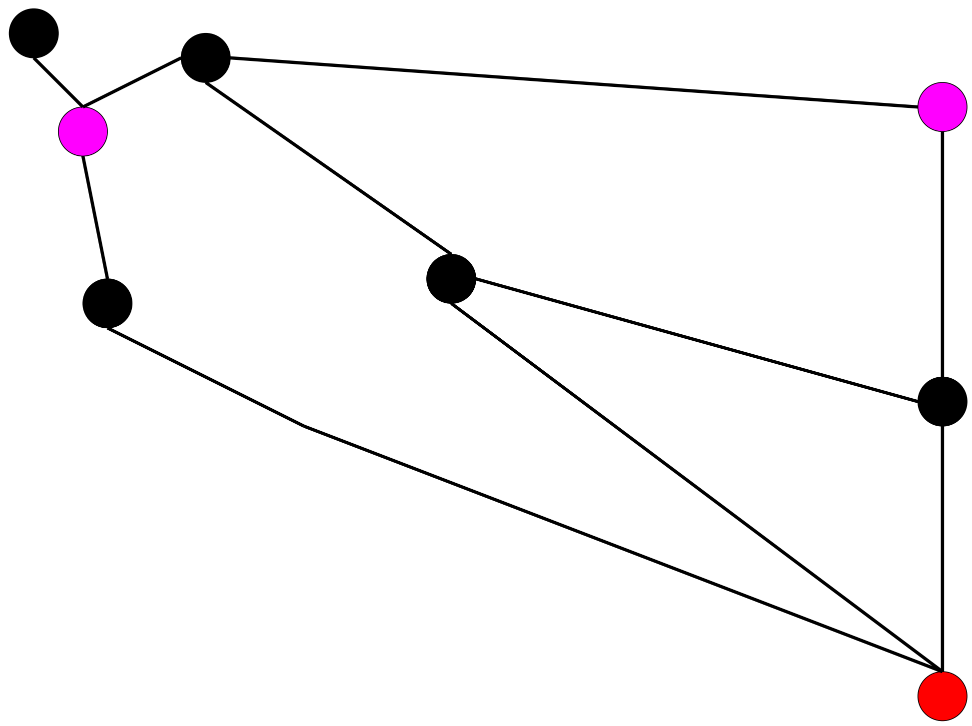
Der reduzierte Graph und eine mögliche Lösung



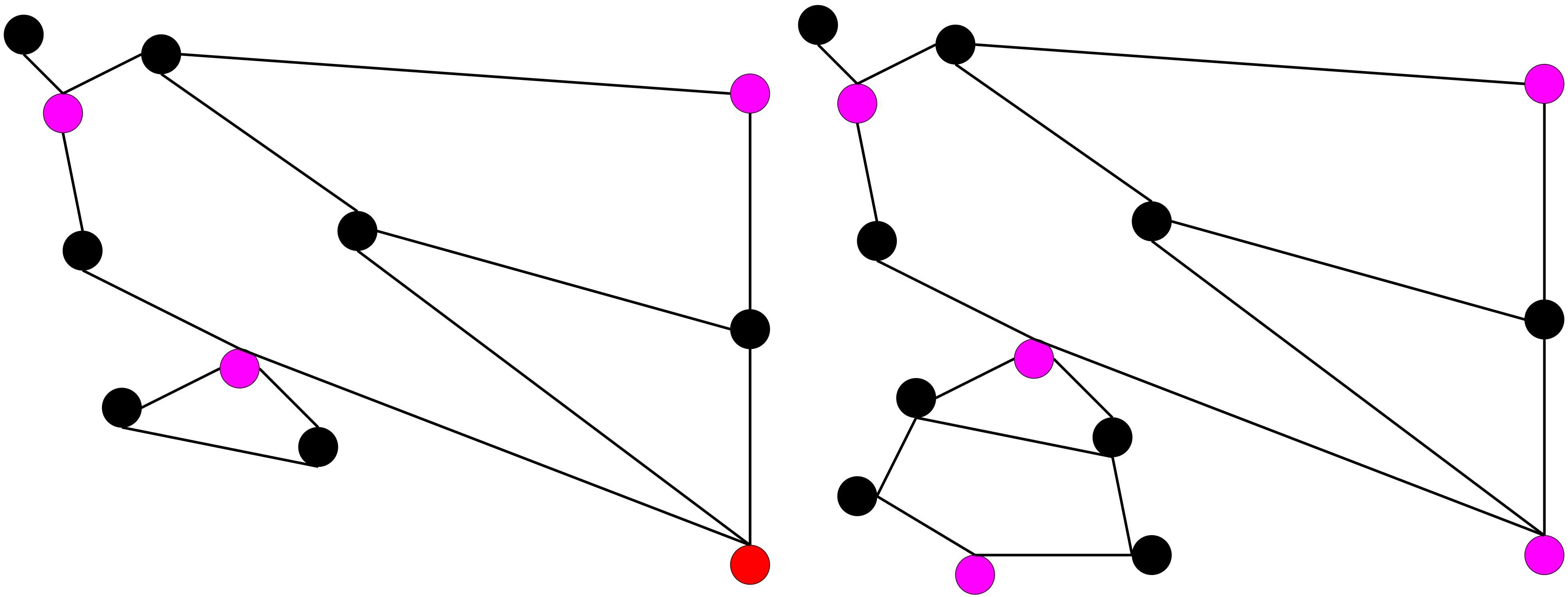


Rückverfolgen der Regelanwendungen. . .

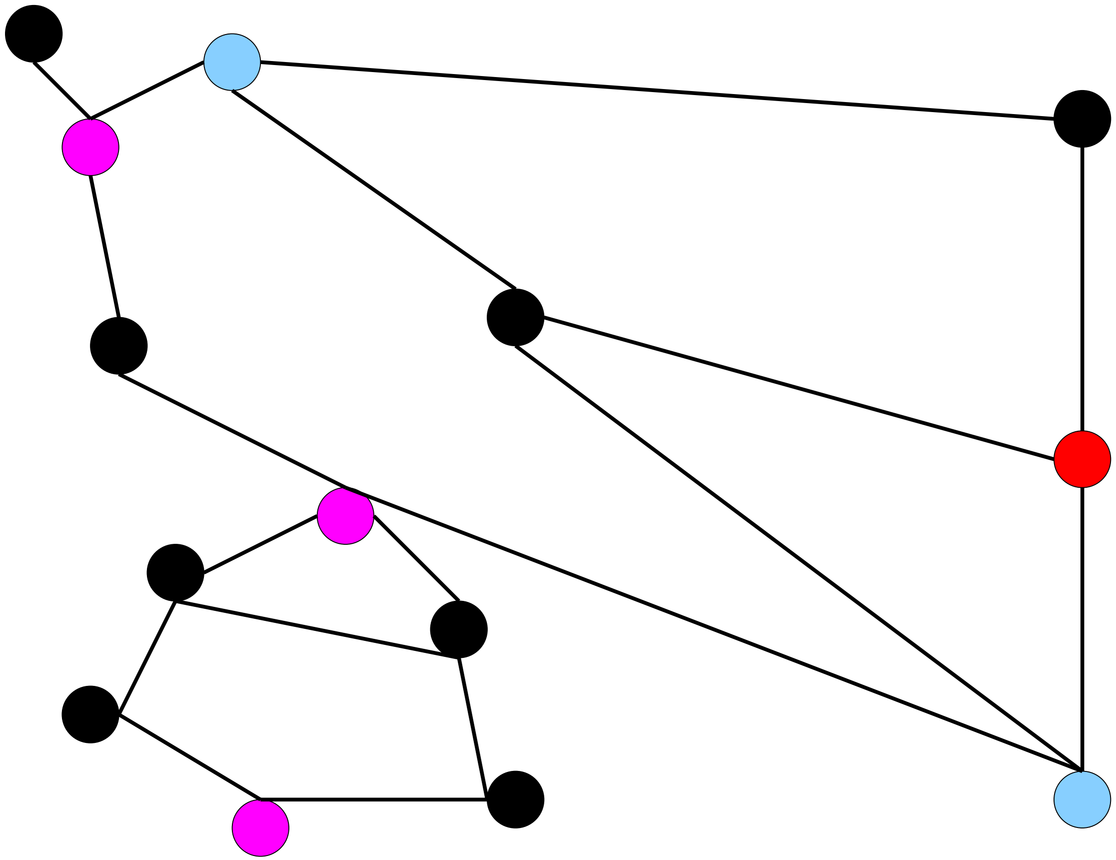




. . . liefert schließlich die Lösung



Diese war nicht optimal wegen falscher Katalysatorwahl



Die Stirling-Formel liefert:

Folgerung 11 NONBLOCKER *kann in Zeit* $\mathcal{O}^*(3.0701^k)$ *gelöst werden.*

Hinweis: Die $\mathcal{O}^*$-Notation "vernachlässigt" sowohl konstante Faktoren als auch polynomielle Anteile und ist daher sehr geeignet für FPT.

Hinweis: Hiermit lässt sich eine (recht) lange als offen geltende Frage leicht beantworten:

> Es gibt einen Algorithmus zum Auffinden kleinster dominierender Mengen in einem Graphen mit n Knoten, dessen Laufzeit sich durch $\mathcal{O}^*(c^n)$ abschätzen lässt für ein $c \in (1, 2)$ unabhängig von n.

Intermezzo

- exakte Exponentialzeitalgorithmen
- siehe: Buch von Fomin / Kratsch aus 2010
- Bei “Mengenauswahlproblemen” (wie DS oder VC) oft trivial: Algorithmus mit Laufzeit $\mathcal{O}^*(2^n)$, wobei n Anzahl der Elemente der Grundmenge (z.B. Knotenmenge): Probiere alle Teilmengen aus.
- Weit weniger trivial: Algorithmus, dessen Laufzeit sich durch $\mathcal{O}^*(c^n)$ abschätzen lässt für ein $c \in (1, 2)$ unabhängig von n
- alternative Sichtweise (gemäß dieser Vorlesung): Parameterisiere mit n.
  Uninteressant für Kernbetrachtungen ...

Kurzer Ausflug: Parameterisierung nach der Knotenzahl

Satz 12 *Es gibt einen Algorithmus zum Auffinden kleinster dominierender Mengen in einem Graphen mit n Knoten, dessen Laufzeit sich durch $\mathcal{O}^*(c^n)$ abschätzen lässt für ein $c \in (1,2)$ unabhängig von n.*

Idee: Unterscheide zwei Fälle:
1. Es gibt keine Nonblocker-Menge der Größe mindestens $k_n > 1/2 \cdot n \rightsquigarrow$
Suche in Zeit $\mathcal{O}^*(3.0701^{k_n})$ mit $3.0701^{k_n} < 2^n$; nach k, $1 \leq k \leq k_n$, sodass k die Größe der größtmöglichen Nonblocker-Menge ist.
2. Es gibt eine Nonblocker-Menge der Größe mindestens $k_n > 1/2 \cdot n$; ist k_n optimal? $\rightsquigarrow$
Probiere alle Mengen der Größe mindestens k_n als Nonblocker-Mengen aus, mit $\binom{n}{k_n} < 2^n$.

Wähle hierbei $k_n = a^{-1}n$ so, dass $3.0701^{k_n} = \binom{n}{k_n} \rightsquigarrow 3.0701 = \frac{a^a}{(a-1)^{(a-1)}} \rightsquigarrow a < 1.9$, $c \approx 1.8$.

Aufgabe: Wie (genau) folgt aus unserem bereits abgeleiteten parameterisierten Algorithmus für VC ein Algorithmus zum Auffinden kleinster Knotenüberdeckungsmengen mit Laufzeit $\mathcal{O}^*(c^n)$ mit $c < 2$? Ermitteln Sie eine gute Laufzeitabschätzung für diesen Algorithmus.

Wie erhalte ich Kernreduktionen? Zusammenfassung

- Fokus auf Extremen (z. B. kleingradige / großgradige Knoten)
  Bsp. Buss-Regel; Nonblocker-Regeln
- Verwendung bekannter (nicht-trivialer) mathematischer Aussagen
  Bsp. Vierfarbensatz; auch NT-Theorem, Blank; McCuaig / Shepherd
- Verallgemeinerung bekannter einfacher Regeln
  Kern für 3-HS?
- Dafür häufig wichtig: Betrachtung annotierter Probleme, z.B. Katalysatoren
- Greedy-Algorithmen für Maximierungsprobleme