Approximation der Cliquenzahl eines Graphen

Matthias Baumgart
matthias.baumgart@informatik.tu-chemnitz.de

Chemnitz, 28. April 2004

Gliederung

1. Einleitung

2. Der Algorithmus von Feige

   1.1. Verständnis
   1.2. Korrektheit
   1.3. Laufzeit

3. Eine Approximationsgüte von $O(n(\log\log n)^2/(\log n)^3)$

Einleitung

Boppana und Halldórsson entwickelten basierend auf der Ramseytheorie einen Algorithmus zur Bestimmung unabhängiger Mengen.

Idee: Ein Graph mit $R(k, l)$ Knoten enthält eine Clique der Größe k oder eine unabhängige Menge der Größe l (oder beides).

Einleitung

Boppana und Halldórsson entwickelten basierend auf der Ramseytheorie einen Algorithmus zur Bestimmung unabhängiger Mengen.

Idee: Ein Graph mit $R(k,l)$ Knoten enthält eine Clique der Größe k oder eine unabhängige Menge der Größe l (oder beides).

Findet der Algorithmus eine große Clique (und nur eine kleine unabhängige Menge), dann entferne diese Clique und suche auf dem Restgraphen weiter nach einer unabhängigen Menge.

Analoges gilt auch für die Approximation der Cliquenzahl, d.h. statt Cliquen müssen nun unabhängige Mengen entfernt werden.

Idee von U. Feige: Entferne nicht nur unabhängige Mengen, sondern auch Untergraphen, welche nur „wenige“ Kanten enthalten.

Idee von U. Feige: Entferne nicht nur unabhängige Mengen, sondern auch Untergraphen, welche nur „wenige“ Kanten enthalten.

Definition 1 *Sei* $G = (V, E)$ *ein Graph mit einer Clique* C *der Kardinalität* $|C| \geq n/k$. *Eine Knotenmenge* S *in* G *heißt schwach, wenn der auf* S *induzierte Untergraph von* G *keine Clique* C_S *der Kardinalität* $|C_S| \geq |S|/(2k)$ *enthält.*

Idee von U. Feige: Entferne nicht nur unabhängige Mengen, sondern auch Untergraphen, welche nur „wenige“ Kanten enthalten.

Definition 1 *Sei* $G = (V, E)$ *ein Graph mit einer Clique* C *der Kardinalität* $|C| \geq n/k$. *Eine Knotenmenge* S *in* G *heißt schwach, wenn der auf* S *induzierte Untergraph von* G *keine Clique* C_S *der Kardinalität* $|C_S| \geq |S|/(2k)$ *enthält.*

Satz 2 *Sei* $G = (V, E)$ *ein Graph mit einer Clique* C *der Kardinalität* $|C| \geq n/k$. *Seien* $S_1, \ldots, S_l$ *beliebige disjunkte schwache Knotenmengen von* G. *Sei* $G' = (V', E')$ *der auf* $V \setminus \{S_1 \cup \ldots \cup S_l\}$ *induzierte Untergraph von* G. *Die Anzahl Knoten in* G' *ist* $|V'| \geq n/(2k)$. *Außerdem enthält* G' *eine Clique* C' *der Kardinalität* $|C'| \geq |V'|/k$.

Beweis:

die Vereinigung $S = \bigcup_{i=1}^{l} S_i$ ist eine schwache Knotenmenge auf höchstens n Knoten und hat demnach keine Clique der Größe $n/(2k)$

nach Voraussetzung hat G eine Clique C der Kardinalität $|C| \geq n/k$

$\Longrightarrow$ mindestens $n/(2k)$ Knoten der Clique C müssen in G' sein

Beweis:

die Vereinigung $S = \bigcup_{i=1}^{l} S_i$ ist eine schwache Knotenmenge auf höchstens n Knoten und hat demnach keine Clique der Größe $n/(2k)$

nach Voraussetzung hat G eine Clique C der Kardinalität $|C| \geq n/k$

$\Longrightarrow$ mindestens $n/(2k)$ Knoten der Clique C müssen in G' sein

Annahme: G' enthält nur Cliquen C' mit $|C'| < |V'|/k$

$\Longrightarrow$ dann befinden sich jedoch mindestens

$$\frac{|V|}{k} - \frac{|V'|}{k} = \frac{|S|}{k}$$

Knoten der Clique C in der Knotenmenge S

$\Longrightarrow$ Widerspruch, da S schwache Knotenmenge ist $\square$

Der Algorithmus von Feige – Verständnis

Der Algorithmus von Feige liefert nach Eingabe eines Graphen $G = (V, E)$, welcher eine Clique der Größe $|V|/k$ enthält, eine Clique C der Kardinalität

$$|C| \geq t \cdot \log_{3k}(|V|/t - 3)$$

für einen Parameter $t \ll n/k$.

Der Algorithmus von Feige – Verständnis

Der Algorithmus von Feige liefert nach Eingabe eines Graphen $G = (V, E)$, welcher eine Clique der Größe $|V|/k$ enthält, eine Clique C der Kardinalität

$$|C| \geq t \cdot \log_{3k}(|V|/t - 3)$$

für einen Parameter $t \ll n/k$.

Einteilung des Algorithmus in Phasen (*Zeilen 3 bis 38*) und Iterationen (*Zeilen 7 bis 32*).

Jede Phase kann aus mehreren einzelnen Iterationen bestehen.

Jede Phase arbeitet auf einem Graphen $G' = (V', E')$, welcher eine Clique der Größe $|V'|/k$ enthält.

Nach Abarbeitung einzelner Iterationen endet eine Phase, so dass eine der folgenden zwei Bedingungen erfüllt ist:

1. Eine Clique C der Kardinalität

$$|C| \geq t \cdot \log_{3k}\left(|V'|/(6kt)\right)$$

   wurde gefunden.

2. Ein schwacher Untergraph $G'' = (V'', E'')$ wurde gefunden.

Jede Iteration arbeit auf einem Graphen $G'' = (V'', E'')$ und führt folgende Schritte aus:

1. Falls $|V''| < 6kt$, dann beende Phase und gib C aus.
2. Partitioniere V'' in disjunkte Knotenmengen P_i der Größe $2kt$.
3. Betrachte alle möglichen t-elementigen Teilmengen S von P_i.
4. Sei $N(S)$ alle Knoten in $V'' \setminus S$, die mit jedem Knoten aus S in G'' verbunden sind. Bezeichne S als *gut*, falls S eine Clique ist und $N(S) \geq |V''|/(2k) - t$ erfüllt.
5. Falls eine *gute* Knotenmenge S gefunden wird, dann setze $C = C \cup S$ und starte neue Iteration mit dem auf $N(S)$ induzierten Untergraphen von G''.
6. Sonst bezeichne V'' als *schwach* und beende Phase.

Der Algorithmus von Feige – Korrektheit

Satz 3 *Wenn eine Knotenmenge* V'' *vom Algorithmus von Feige als schwach erkannt wird, dann enthält der auf* V'' *induzierte Untergraph von* $G = (V, E)$ *tatsächlich keine Clique* C *der Kardinalität*

$$|C| \geq \frac{|V''|}{2k}\ .$$

Der Algorithmus von Feige – Korrektheit

Satz 3 *Wenn eine Knotenmenge V'' vom Algorithmus von Feige als schwach erkannt wird, dann enthält der auf V'' induzierte Untergraph von $G = (V, E)$ tatsächlich keine Clique C der Kardinalität*

$$|C| \geq \frac{|V''|}{2k} \,.$$

Beweis: *Annahme:* $V'' = P_1 \cup \ldots \cup P_l$ enthält Clique C mit

$$|C| \geq \frac{|V''|}{2k} = t \cdot l$$

nach Schubfachprinzip gibt es ein P_i, dass t Knoten von C enthält

$\Longrightarrow$ es gibt ein S mit der Eigenschaft *gut* $\square$

Satz 4 *Endet eine Phase mit der Ausgabe einer Menge C, dann enthält diese mindestens*

$$|C| \geq t \cdot \log_{3k} \frac{|V'|}{6kt}$$

Knoten, welche eine Clique in $G' = (V', E')$ und damit auch in $G = (V, E)$ bilden.

Satz 4 *Endet eine Phase mit der Ausgabe einer Menge C, dann enthält diese mindestens*

$$|C| \geq t \cdot \log_{3k} \frac{|V'|}{6kt}$$

Knoten, welche eine Clique in $G' = (V', E')$ und damit auch in $G = (V, E)$ bilden.

Beweis:

jede – außer der letzten – Iteration fügt t Knoten zu C hinzu

$\Longrightarrow$ Bestimmung einer unteren Schranke für die Anzahl Iterationen

- die erste Iteration startet mit V' vielen Knoten
- eine neue Iteration startet mit wenigstens $|V''|/(2k) - t$
- es gilt $|V''| \geq 6kt$, also $t \leq |V''|/(6k)$
- die Anzahl Knoten einer neuen Iteration ist mindestens

$$\frac{|V''|}{2k} - \frac{|V''|}{6k} = \frac{|V''|}{3k}$$

- die Anzahl Knoten der $i+1$'ten Iteration ist mindestens

$$\frac{|V'|}{(3k)^i}$$

$\Longrightarrow$ Wann ist die Anzahl Knoten einer Iteration kleiner $6kt$?

$$\begin{aligned} & \frac{|V'|}{(3k)^x} &<& \quad 6kt \\ \iff & \frac{|V'|}{6kt} &<& \quad (3k)^x \\ \iff & \log_{3k} \frac{|V'|}{6kt} &<& \quad x \end{aligned}$$

$\Longrightarrow$ Wann ist die Anzahl Knoten einer Iteration kleiner $6kt$?

$$\begin{aligned} \frac{|V'|}{(3k)^x} &< 6kt \\ \iff \quad \frac{|V'|}{6kt} &< (3k)^x \\ \iff \quad \log_{3k} \frac{|V'|}{6kt} &< x \end{aligned}$$

es werden mindestens $\log_{3k}(|V'|/(6kt))$ Iterationen durchlaufen, die jeweils t Knoten zu C hinzufügen, also gilt

$$|C| \geq t \cdot \log_{3k} \frac{|V'|}{6kt} \qquad \square$$

Der Algorithmus von Feige – Laufzeit

$\Longrightarrow$ mehrere Phasen gibt es nur, wenn schwache Knotenmengen gefunden werden

- jede schwache Knotenmenge S erfüllt $|S| \geq 6kt$, also gilt

$$\# \text{ Phasen} \leq \frac{n}{6kt}$$

- die Anzahl Iteration einer Phase ist beschränkt durch

$$\# \text{ Iterationen} \leq \frac{n}{t}$$

$\Longrightarrow$ Laufzeit einer Iteration

- die Anzahl Knotenmengen P_i ist beschränkt durch

$$\#\ \text{Knotenmengen}\ P_i \leq \frac{n}{2kt}$$

- Anzahl möglicher Mengen S

$$\#\ t\text{-elementige Knotenmengen von}\ P_i = \binom{2kt}{t}$$

- Test auf Eigenschaft *gut* in Zeit $O(n \cdot t)$

$\Longrightarrow$ Laufzeit einer Iteration

- die Anzahl Knotenmengen P_i ist beschränkt durch

$$\#\ \text{Knotenmengen}\ P_i \leq \frac{n}{2kt}$$

- Anzahl möglicher Mengen S

$$\#\ t\text{-elementige Knotenmengen von}\ P_i = \binom{2kt}{t}$$

- Test auf Eigenschaft *gut* in Zeit $O(n \cdot t)$

$$\Longrightarrow\ \text{Gesamtlaufzeit}\ O\left(\binom{2kt}{t} \cdot \frac{n^4}{k^2 \cdot t^2}\right)$$

Laufzeit polynomiell in n, wenn $\binom{2kt}{t}$ polynomiell in n

die Wahl des Parameters t beeinflußt Größe der gefundenen Clique sowie die Laufzeit des Algorithmus

Laufzeit polynomiell in n, wenn $\binom{2kt}{t}$ polynomiell in n

die Wahl des Parameters t beeinflußt Größe der gefundenen Clique sowie die Laufzeit des Algorithmus

um beides miteinander abzustimmen und eine polynomiell in n beschränkte Laufzeit zu erhalten, setzt man

$$t = \Theta\left(\frac{\log n}{\log \log n}\right)$$

Laufzeit polynomiell in n, wenn $\binom{2kt}{t}$ polynomiell in n

die Wahl des Parameters t beeinflußt Größe der gefundenen Clique sowie die Laufzeit des Algorithmus

um beides miteinander abzustimmen und eine polynomiell in n beschränkte Laufzeit zu erhalten, setzt man

$$t = \Theta\left(\frac{\log n}{\log \log n}\right)$$

die gefundene Clique C hat dann die Kardinalität

$$|C| = \Omega\left(\left(\frac{\log n}{\log \log n}\right)^2\right)$$

Eine Approximationsgüte von $O(n(\log\log n)^2/(\log n)^3)$

Gegeben: ein Graph $G = (V, E)$ mit einer Clique C der Kardinalität

$$|C| \geq \frac{n}{k} \qquad (k \text{ minimal})$$

Eine Approximationsgüte von $O(n(\log\log n)^2/(\log n)^3)$

Gegeben: ein Graph $G = (V, E)$ mit einer Clique C der Kardinalität

$$|C| \geq \frac{n}{k} \qquad (k \text{ minimal})$$

für k kommen nur folgende n Werte in Frage:

$$k \in \left\{\frac{n}{1}, \frac{n}{2}, \ldots, \frac{n}{n-1}, 1\right\}$$

o.B.d.A. sei k bekannt

Fall 1: $k \geq (\log n)^3$

- durch Ausgabe eines beliebigen Knotens $v \in V$ erreicht man eine Approximationsgüte von $O(n/(\log n)^3)$

Fall 1: $k \geq (\log n)^3$

- durch Ausgabe eines beliebigen Knotens $v \in V$ erreicht man eine Approximationsgüte von $O(n/(\log n)^3)$

Fall 2: $k \leq \log n/(2\log\log n)$

- benutze Algorithmus *IndependentSetRemoval* von Boppana und Halldórsson
- bei einem Graphen G, welcher eine Clique der Kardinalität $2n\log\log n/\log n$ enthält, liefert dieser eine Clique der Größe $(\log n)^3/(6\log\log n)$
- Approximationsgüte von $O(n\log\log n/(\log n)^3)$

Fall 3: $\log n/(2\log\log n) < k < (\log n)^3$

- der Algorithmus von Feige erreicht hier nur eine Approximationsgüte von $O(n(\log\log n)^3/(\log n)^3)$

- für $k > \log n$ ist die Güte schon $O(n(\log\log n)^2/(\log n)^3)$

- müssen also einen Faktor von $\Omega(\log\log n)$ einsparen

Fall 3: $\log n/(2\log\log n) < k < (\log n)^3$

- der Algorithmus von Feige erreicht hier nur eine Approximationsgüte von $O(n(\log\log n)^3/(\log n)^3)$

- für $k > \log n$ ist die Güte schon $O(n(\log\log n)^2/(\log n)^3)$

- müssen also einen Faktor von $\Omega(\log\log n)$ einsparen

$\Longrightarrow$ modifiziere Algorithmus von Feige

$\Longrightarrow$ ändere Definition einer *guten* Knotenmenge

Modifikation:

- bezeichne eine Knotenmenge S als *gut*, falls S eine Clique ist und $N(S) > n_{test} - t$ gilt, wobei n_{test} der größte Wert ist für den gilt

$$n_{test} \leq \frac{\log n_{test}}{2\log\log n_{test}} \cdot \frac{|V''|}{2k}$$

Modifikation:

- bezeichne eine Knotenmenge S als *gut*, falls S eine Clique ist und $N(S) > n_{test} - t$ gilt, wobei n_{test} der größte Wert ist für den gilt

$$n_{test} \leq \frac{\log n_{test}}{2\log\log n_{test}} \cdot \frac{|V''|}{2k}$$

- falls $|V''|/(2k) - t \leq N(S) \leq n_{test} - t$ gilt, dann führe den Algorithmus *IndependentSetRemoval* auf $G[S \cup N(S)]$ aus

Modifikation:

- bezeichne eine Knotenmenge S als *gut*, falls S eine Clique ist und $N(S) > n_{test} - t$ gilt, wobei n_{test} der größte Wert ist für den gilt

$$n_{test} \leq \frac{\log n_{test}}{2\log\log n_{test}} \cdot \frac{|V''|}{2k}$$

- falls $|V''|/(2k) - t \leq N(S) \leq n_{test} - t$ gilt, dann führe den Algorithmus *IndependentSetRemoval* auf $G[S \cup N(S)]$ aus

  $\Longrightarrow$ erhält man eine Clique der Größe $(\log n_{test})^3/(6\log\log n_{test})$, dann fügt man diese Clique zu C hinzu und gibt C aus

  $\Longrightarrow$ sonst ist S eine *schwache* Knotenmenge

Modifikation:

- bezeichne eine Knotenmenge S als *gut*, falls S eine Clique ist und $N(S) > n_{test} - t$ gilt, wobei n_{test} der größte Wert ist für den gilt

$$n_{test} \leq \frac{\log n_{test}}{2\log\log n_{test}} \cdot \frac{|V''|}{2k}$$

- falls $|V''|/(2k) - t \leq N(S) \leq n_{test} - t$ gilt, dann führe den Algorithmus *IndependentSetRemoval* auf $G[S \cup N(S)]$ aus

  $\Longrightarrow$ erhält man eine Clique der Größe $(\log n_{test})^3/(6\log\log n_{test})$, dann fügt man diese Clique zu C hinzu und gibt C aus

  $\Longrightarrow$ sonst ist S eine *schwache* Knotenmenge

Korrektheit des modifizierten Algorithmus muss gezeigt werden

Satz 5 *Wenn eine Knotenmenge V'' vom modifizierten Algorithmus von Feige als schwach erkannt wird, dann enthält der auf V'' induzierte Untergraph von $G = (V, E)$ tatsächlich keine Clique C der Kardinalität $|C| \geq |V''|/(2k)$.*

Satz 5 *Wenn eine Knotenmenge V'' vom modifizierten Algorithmus von Feige als schwach erkannt wird, dann enthält der auf V'' induzierte Untergraph von $G = (V, E)$ tatsächlich keine Clique C der Kardinalität $|C| \geq |V''|/(2k)$.*

Beweis:

- zunächst analog zum Beweis von Satz 3

- angenommen $G[V'']$ enthält eine Clique der Größe $|V''|/(2k)$, dann existiert nach dem Schubfachprinzip eine Knotenmenge P_i, die mindestens t Knoten von dieser Clique enthält

- dann gibt es aber eine t-elementige Menge S, welche eine Clique ist und für die $|N(S)| \geq |V''|/(2k) - t$ gilt

$\Longrightarrow$ gilt zusätzlich $|N(S)| > n_{test} - t$, dann ist S nicht *schwach*

$\Longrightarrow$ sonst gilt $|V''|/(2k) - t \leq N(S) \leq n_{test} - t$

$\Longrightarrow$ gilt zusätzlich $|N(S)| > n_{test} - t$, dann ist S nicht *schwach*

$\Longrightarrow$ sonst gilt $|V''|/(2k) - t \leq N(S) \leq n_{test} - t$

- der Graph $G[S \cup N(S)]$ enthält dann eine Clique C der Größe

$$|C| \geq \frac{|V''|}{2k} = \frac{n_{test} \cdot 2\log\log n_{test}}{\log n_{test}}$$

- der Algorithmus *IndependentSetRemoval* liefert dann eine Clique der Kardinalität

$$|C| \geq \frac{(\log n_{test})^3}{6\log\log n_{test}}$$

$\Longrightarrow$ gilt zusätzlich $|N(S)| > n_{test} - t$, dann ist S nicht *schwach*

$\Longrightarrow$ sonst gilt $|V''|/(2k) - t \leq N(S) \leq n_{test} - t$

- der Graph $G[S \cup N(S)]$ enthält dann eine Clique C der Größe

$$|C| \geq \frac{|V''|}{2k} = \frac{n_{test} \cdot 2 \log\log n_{test}}{\log n_{test}}$$

- der Algorithmus *IndependentSetRemoval* liefert dann eine Clique der Kardinalität

$$|C| \geq \frac{(\log n_{test})^3}{6 \log\log n_{test}}$$

$\Longrightarrow$ V'' wird nicht als *schwach* bezeichnet $\square$

Satz 6 *Es seien* $t = \log n/\log\log n$ *und* $\log n/(2\log\log n) < k < \log n$ *die Parameter für den modifizierten Algorithmus von Feige. Endet eine Phase dieses Algorithmus mit der Ausgabe einer Menge* C*, dann ist* C *eine Clique auf* $\Omega(t \cdot \log_b |V'|)$ *Knoten, wobei* $b = \Theta(k \cdot \log\log |V'|/\log |V'|)$*.*

Satz 6 *Es seien* $t = \log n/\log\log n$ *und* $\log n/(2\log\log n) < k < \log n$ *die Parameter für den modifizierten Algorithmus von Feige. Endet eine Phase dieses Algorithmus mit der Ausgabe einer Menge* C*, dann ist* C *eine Clique auf* $\Omega(t \cdot \log_b |V'|)$ *Knoten, wobei* $b = \Theta(k \cdot \log\log |V'|/\log |V'|)$*.*

Insbesondere bedeutet dies, dass eine Clique C der Kardinalität

$$|C| = \Omega\left(\frac{(\log n)^2}{\log\log n}\right)$$

gefunden wird, wenn der Graph eine Clique der Größe $\Theta(n\log\log n/\log n)$ hat.

Beweis:

$\Longrightarrow$ betrachte nur Iteration solange $|V''| \geq \sqrt{|V'|}$ gilt

- findet der Algorithmus *IndependentSetRemoval* eine Clique C der Kardinalität

$$|C| \geq \frac{(\log n_{test})^3}{6 \log\log n_{test}} \ ,$$

dann gilt wegen $n_{test} \geq |V''|/(2k) \geq \sqrt{|V'|}/(2k)$

$$|C| = \Omega\left(\frac{(\log |V'|)^3}{\log\log |V'|}\right)$$

- sonst gilt in jeder Iteration $|N(S)| > n_{test} - t$

- sonst gilt in jeder Iteration $|N(S)| > n_{test} - t$
- die Anzahl Knoten einer neuen Iteration ist dann

$$\Theta\left(\frac{|V''| \cdot \log|V'|}{k \cdot \log\log|V'|}\right)$$

- die Reduzierung der Knotenmenge V'' erfolgt dann mit einem Faktor von

$$O\left(\frac{k \cdot \log\log|V'|}{\log|V'|}\right)$$

(vorher war dieser Faktor $O(k)$)

- die Anzahl Iterationen ist dann

$$\begin{aligned} \left(\frac{\log|V'|}{k\cdot\log\log|V'|}\right)^x\cdot|V'| &< 6kt \\ \iff \qquad \frac{\log(|V'|/(6kt))}{\log(\underbrace{k\cdot\log\log|V'|/\log|V'|}_{=b})} &< x \end{aligned}$$

- die Anzahl Iterationen ist dann

$$\begin{aligned} \left(\frac{\log|V'|}{k\cdot\log\log|V'|}\right)^x \cdot |V'| &< 6kt \\ \iff \qquad \frac{\log(|V'|/(6kt))}{\log(\underbrace{k\cdot\log\log|V'|/\log|V'|}_{=b})} &< x \end{aligned}$$

- in jeder Iteration werden t Knoten zu C hinzugefügt

$\Longrightarrow$ man erhält eine Clique C der Kardinalität

$$|C| = \Omega(t\cdot\log_b|V'|) \qquad \square$$