Algorithmen II
Vorlesung am 22.11.2012

Randomisierte Algorithmen – Max Cut

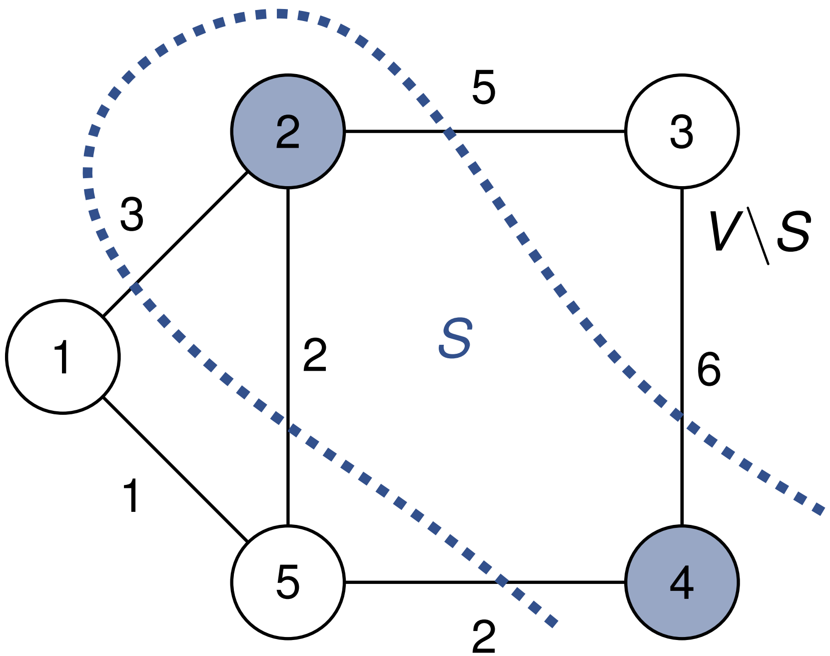
INSTITUT FÜR THEORETISCHE INFORMATIK · PROF. DR. DOROTHEA WAGNER

Problemdefinition

Problem MaxCut: Gegeben ist ein Graph $G = (V, E)$ mit Gewichtsfunktion $c\colon E \longrightarrow \mathbb{N}$. Gesucht ist ein Schnitt $(S, V \setminus S)$ von G mit maximalem Gewicht, d.h.

$$c(S, V \setminus S) := \sum_{\substack{u,v \in E \\ u \in S \text{ und } v \in V \setminus S}} c(\{u, v\}) \qquad \text{soll maximal sein.}$$

Problem ist $\mathcal{NP}$-schwer.



$c(S, V \setminus S) = 18$

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Problemdefinition

Problem MAXCUT: Gegeben ist ein Graph $G = (V, E)$ mit Gewichtsfunktion $c\colon E \longrightarrow \mathbb{N}$. Gesucht ist ein Schnitt $(S, V \setminus S)$ von G mit maximalem Gewicht, d.h.

$$c(S, V \setminus S) := \sum_{\substack{u,v\,\in\, E \\ u\in S \text{ und } v\in V\setminus S}} c(\{u, v\}) \qquad \text{soll maximal sein.}$$

Problem ist $\mathcal{NP}$-schwer.

$G = (V, E)$ $\xrightarrow{\text{Transformation}}$ ganzzahliges quadratisches Programm (IQP) über $\mathbb{N}$ $\xrightarrow{\text{Relaxierung}}$ reellwertiges quadratisches Programm (QP^2) über $\mathbb{R}^2$

Lösung $\xleftarrow{\text{Transformation mithilfe eines randomisierten Algorithmus}}$ reellwertiges quadratisches Programm (QP^2) über $\mathbb{R}^2$

Ganzzahliges Quadratisches Programm

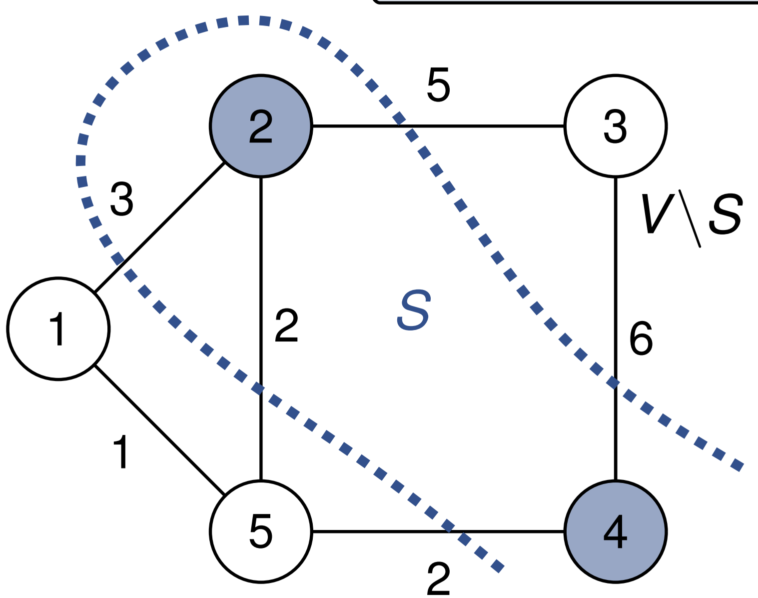
Zu i und $j \in V := \{1, \ldots, n\}$ definiere $c_{ij} := \begin{cases} c(\{i,j\}) & \text{falls } \{i,j\} \in E \\ 0 & \text{sonst} \end{cases}$

IQP(I):

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - x_i \cdot x_j)$$

unter den Nebenbedingungen

$$x_i, x_j \in \{-1, 1\} \quad \text{und} \quad 1 \leq i, j \leq n \ .$$



Gewichtungsmatrix:

	1	2	3	4	5
1		3			1
2	3		5		2
3		5	6		
4			6		2
5	1	2		2	

Lösung:

$x_2 = x_4 = 1$

$x_1 = x_5 = x_3 = -1$

Relaxierung von IQP(I)

Sei $x^i = (x^i_1, x^i_2) \in \mathbb{R}^2$ normierter Vektor.

QP$^2(I)$:

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - x^i \cdot x^j)$$

unter den Nebenbedingungen

$$x^i, x^j \in \mathbb{R}^2 \text{ sind normierte Vektoren}$$

Für das Produkt von x^i und x^j gilt:

$$x^i \cdot x^j = x^i_1 \cdot x^j_1 + x^i_2 \cdot x^j_2$$

Beachte: Jede Lösung $(x_1, \ldots, x_n)$ von $IQP(I)$ induziert eine Lösung $(x^1, \ldots, x^n)$ von QP$^2(I)$ mittels $x^i = (x_i, 0)$.

$\Rightarrow$ **QP**^{2}**(I) ist Relaxierung von IQP(I)**

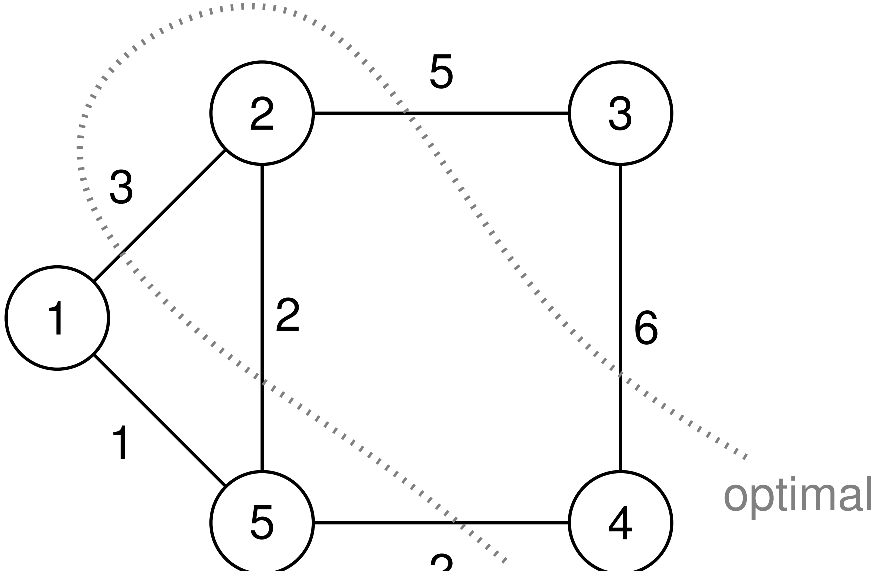
Random MaxCut

Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \underbrace{\widetilde{x^i} \cdot r \geq 0}\}$

$\rightarrow$ $\widetilde{x_i}$ liegt oberhalb der zu r senkrechten Linie ℓ



Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

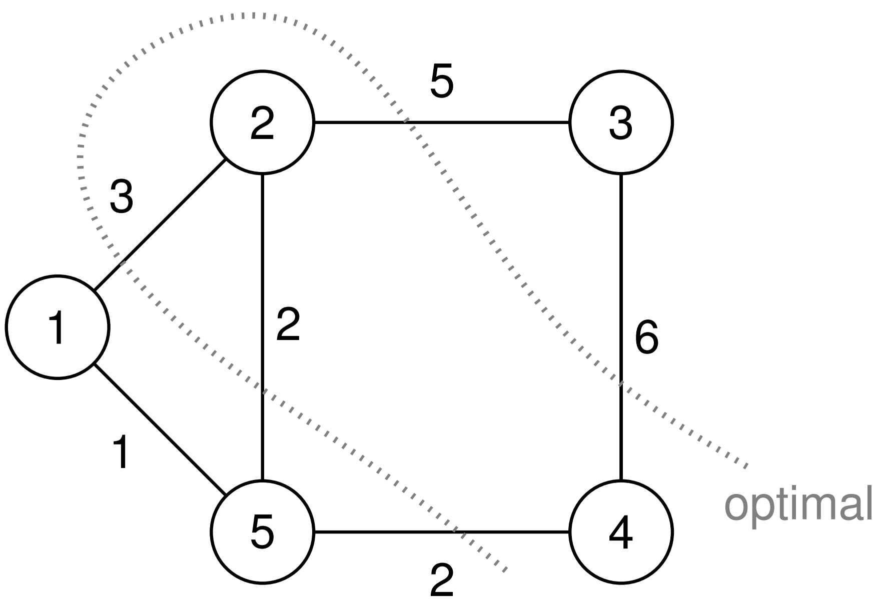
Random MaxCut

Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

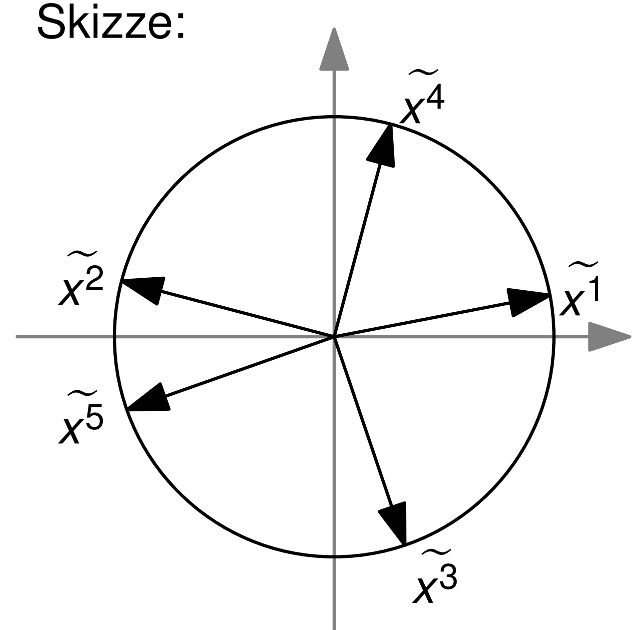
1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \widetilde{x^i} \cdot r \geq 0\}$

$\widetilde{x_i}$ liegt oberhalb der zu r senkrechten Linie ℓ



1. Schritt: Berechne Lösung für QP^2:

Skizze:



Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner

Random MaxCut

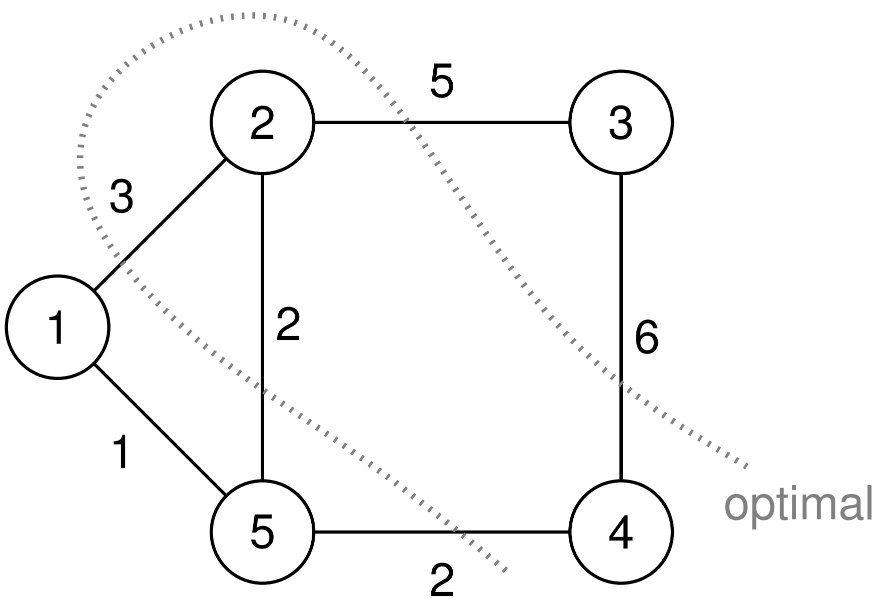
Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

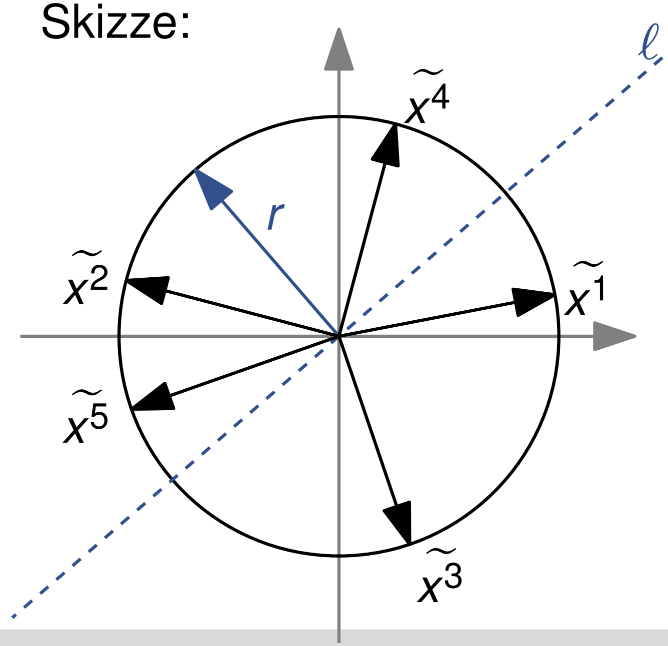
1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \underbrace{\widetilde{x^i} \cdot r \geq 0}\}$

$\rightarrow$ $\widetilde{x_i}$ liegt oberhalb der zu r senkrechten Linie ℓ

2. Schritt: Rate Vektor r



Skizze:



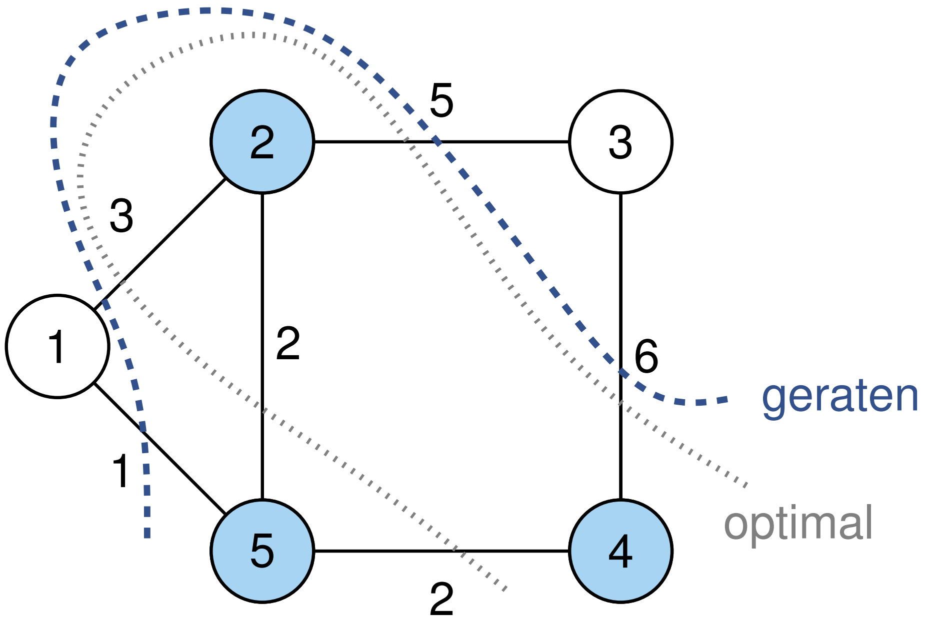
Random MaxCut

Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

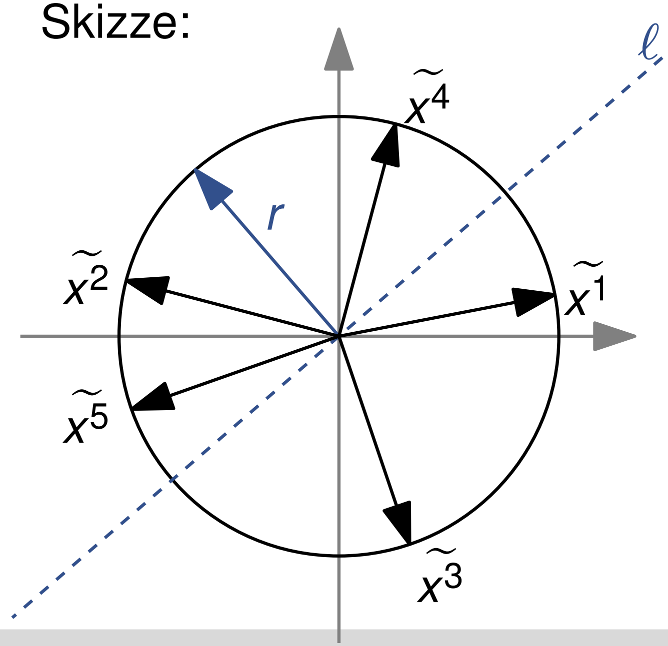
1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \widetilde{x^i} \cdot r \geq 0\}$

$\longrightarrow$ $\widetilde{x_i}$ liegt oberhalb der zu r senkrechten Linie ℓ



3. Schritt: Berechne S

Skizze:



Random MaxCut

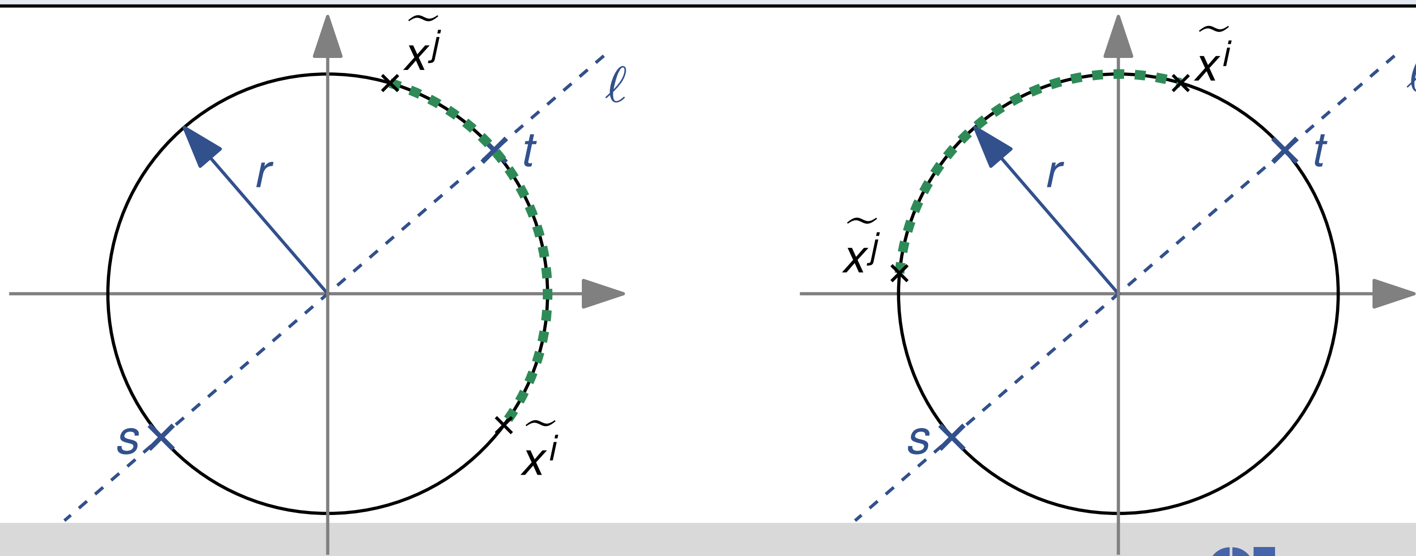
Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \widetilde{x^i} \cdot r \geq 0\}$

Satz 8.18: Sei I eine Instanz für MaxCut und $C_{RMC}(I)$ der Wert der Lösung, die Random MaxCut für I berechnet. Wenn die Vektoren r gleichverteilt angenommen werden, so gilt:

$$E[C_{RMC}(I)] = \frac{1}{\pi} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot \arccos(\widetilde{x^i} \cdot \widetilde{x^j})$$



Random MaxCut

Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \widetilde{x^i} \cdot r \geq 0\}$

Satz 8.19: Für eine Instanz I von MaxCut berechnet Random MaxCut eine Lösung mit dem Wert $C_{\text{RMC}}(I)$, für die gilt

$$\frac{E[C_{\text{RMC}}(I)]}{OPT(I)} \geq 0,8785 \ .$$

Random MaxCut

Eingabe: Graph $G = (V, E)$ mit einer Gewichtsfunktion $c : E \longrightarrow \mathbb{N}$

Ausgabe: Ein Schnitt $(S, V \setminus S)$ in G

1. Berechne optimale Lösung $(\widetilde{x^1}, \ldots, \widetilde{x^n})$ für QP^2(G,c)
2. Wähle zufällig einen zweidimensionalen Vektor r mit Norm 1
3. $S \leftarrow \{i \in V : \widetilde{x^i} \cdot r \geq 0\}$

- RANDOM MAXCUT ist polynomiell, falls Schritt 1 polynomiell ist.
- Bisher nicht bekannt, ob QP^2 polynomiell gelöst werden kann.
- Idee: Modifiziere QP^2 so, dass RANDOM MAXCUT polynomiell gelöst wird.

Effiziente Lösung von $QP^2(I)$

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - x^i \cdot x^j)$$

unter den Nebenbedingungen

$x^i, x^j \in \mathbb{R}^2$ sind normierte Vektoren.

ist äquivalent zu

QP^n

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - m_{ij})$$

mit $m_{ij} = x^i \cdot x^j$ und $m_{ii} = 1$ und unter den Nebenbedingungen

$x^i, x^j \in \mathbb{R}^n$ sind normierte Vektoren.

Effiziente Lösung von $QP^2(I)$

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - x^i \cdot x^j)$$

unter den Nebenbedingungen

$x^i, x^j \in \mathbb{R}^2$ sind normierte Vektoren.

ist äquivalent zu

QP^n

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - m_{ij})$$

mit $m_{ij} = x^i \cdot x^j$ und $m_{ii} = 1$ und unter den Nebenbedingungen

$x^i, x^j \in \mathbb{R}^n$ sind normierte Vektoren.

Definition 8.20: Eine $n \times n$-Matrix M heißt *positiv semidefinit*, falls für jeden Vektor $x \in \mathbb{R}^n$ gilt:

$$x^T \cdot M \cdot x \geq 0$$

Die Matrix $M := (m_{ij})$ ist positiv semidefinit.

Denn: Symmetrische Matrix M ist genau dann positiv semidefinit, wenn es $m \times n$-Matrix P ($m \leq n$) gibt, sodass

$$M = P^T \cdot P$$

(P ist in polynomieller Zeit berechenbar, falls M positiv semidefinit.)

(Für jede positiv semidefinite $n \times n$-Matrix M mit $m_{ii} = 1$ gilt, dass n normierte Vektoren $x^1, \ldots, x^n \in \mathbb{R}^n$ mit $m_{ij} = x^i \cdot x^j$ in polynomieller Zeit berechnet werden können.)

Effiziente Lösung von $QP^2(I)$

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - x^i \cdot x^j)$$

unter den Nebenbedingungen

$x^i, x^j \in \mathbb{R}^2$ sind normierte Vektoren.

ist äquivalent zu

QP^n

$$\max \frac{1}{2} \cdot \sum_{j=1}^{n} \sum_{i=1}^{j-1} c_{ij} \cdot (1 - m_{ij})$$

mit $m_{ij} = x^i \cdot x^j$ und $m_{ii} = 1$ und unter den Nebenbedingungen

$x^i, x^j \in \mathbb{R}^n$ sind normierte Vektoren.

Definition 8.20: Eine $n \times n$-Matrix M heißt *positiv semidefinit*, falls für jeden Vektor $x \in \mathbb{R}^n$ gilt:

$$x^T \cdot M \cdot x \geq 0$$

Die Matrix $M := (m_{ij})$ ist positiv semidefinit.

Damit entspricht $QP^n(I)$ dem Problem SEMI-DEFINIT-CUT(I).

Satz: Es gibt Algorithmus $\mathcal{A}_\epsilon$ für jedes ϵ, sodass $\mathcal{A}_\epsilon$ polynomiell in Eingabegröße und $\log(\frac{1}{\epsilon})$ ist und

$$\mathcal{A}_\epsilon(I) \geq OPT_{SD}(I) - \epsilon,$$

wobei $OPT_{SD}(I)$ optimaler Lösungswert von SEMI-DEFINIT-CUT(I).

(Ohne Beweis)

Man kann zeigen: Für $\epsilon = 10^{-5}$ wird Approximationsgüte von RANDOM MAXCUT erreicht.

Institut für Theoretische Informatik
Prof. Dr. Dorothea Wagner