Übungsblatt 6

Ali Bektas 588063 Julian Kremer 562717
Ruben Dorfner 550204

11. Dezember 2019

Aufgabe 31

a)

Wir reduzieren VC auf DOMSET.

1. Sei $G = (V, E)$ein **ungerichteter** Graph und sei $G' = (V, E')$ ein **gerichteter** Graph.
2. Für jedes $a, b \in V$ mit $(a, b) \in E$ enthalte $E' \cap = \{(a, b), (b, a)\}$
3. Für jeden Knoten a für den es gilt : $\neg\exists(a, b) \in E$ enthalte $E' \cap = \{(b, a)\}$ für alle $b \in V/\{a\}$

$$\begin{aligned} A \in VC(G,k) &\iff \\ &\iff \text{Für alle e} \in \text{E gilt}, e \cap A \neq \emptyset \\ &\iff \text{Für alle v} \in V'n\text{B gilt} : \exists b \in B : (b, v) \in E' \\ &\iff B \in DOMSET(G', k) \end{aligned}$$

b)

In einem Turniergraph mit n Knoten gibt es $\frac{n\cdot(n-1)}{2}$ Kanten. Im Durchschnitt gibt es also $\frac{(n-1)}{2}$ ausgehende Kanten für jede Knoten. Wir sammeln die DOMSET von einem Graphen wie folgt.

1. Wir nehmen den Knoten , der mindestens $\frac{n-1}{2}$ ausgehende Kanten hat.
2. Wir nehmen ihn und alle von ihm dominierten Knoten heraus.
3. Wir wiederholen Schritt 1 für den übriggebliebenen Graphen.

Wir nehmen jedes Mal $\underbrace{1}_{\text{gewählter Knoten}} + \left\lceil \frac{(n-1)}{2} \right\rceil$ viele Knoten heraus.

Da $n-(1+\left\lceil \frac{(n-1)}{2} \right\rceil) \leq n/2$ wird die Anzahl der Knoten in jedem Schritt mindestens halbiert. Die maximale Anzahl an notwendigen Halbierungsschritten x für einen Turniergraphen der Größe n ergibt sich durch umformen der Gleichung $n \cdot \frac{1}{2}^x = 1$, da der Algorithmus spätestens bei einem Graphen der Größe 1 terminiert. Da $x = log(n)$ gilt impliziert dies, dass das für jeden Turniergraphen ein Dominating Set der Mächtigkeit log(n) existiert.

c)

Da wir wissen , dass wir maximal log(n) Elemente brauchen , um ein Dominating Set für den Graphen zu erstellen ,sei M eine DTM die folgende Arbeitsweise hat:

1. M wählt in aufsteigender Reihenfolge maximal $log(n)$ viele Elemente aus n vielen Elementen
2. Dann versucht sie nachzuweisen dass die gewählte Menge eine Dominating Set für den Grpahen darstellt. Wenn das ihr gelingt hält sie , wenn nicht geht sie weiter.

Somit ist die Anzahl an Rechenschritten :

$$\begin{aligned}
\sum_{i=1}^{\lceil log(n) \rceil} \binom{n}{i} &= \frac{n!}{(n-1)!} + \frac{n!}{(n-2)!2!} + \cdots + \frac{n!}{(n-\lceil log(n) \rceil)! \cdot \lceil log(n) \rceil} \\
&= n + \frac{n*(n-1)}{2} + \cdots + \frac{n \cdot (n-1) \cdot \cdots \cdot (n - \lceil logn \rceil + 1)}{1 \cdot 2 \ldots \lceil logn \rceil} \\
&\leq n + n \cdot (n-1) + \cdots + n \cdot (n-1) \ldots (n - \lceil logn \rceil + 1) = n^{O(logn)}
\end{aligned}$$

Die letzte Schranke setzt hingegen $n \geq 2$ voraus. Es ist aber sinnvoll das anzunehmen. Für $n = 1$ sind wir sowieso in O(1) fertig.