

Der linearzeit MST-Algorithmus

Der schnellste Algorithmus für das MST/ MSF Problem

Max Springenberg

Proseminar: Randomisierte Algorithmen, TU Dortmund

Motivation

„the fastest“

Borůvka, Kruskal, Prim	$O(m \log(n))$	(deterministisch)
Chazelle	$O(m \log(\beta(m, n)))$	(deterministisch)
<i>MST</i>	$O(m + n)$	(randomisiert)

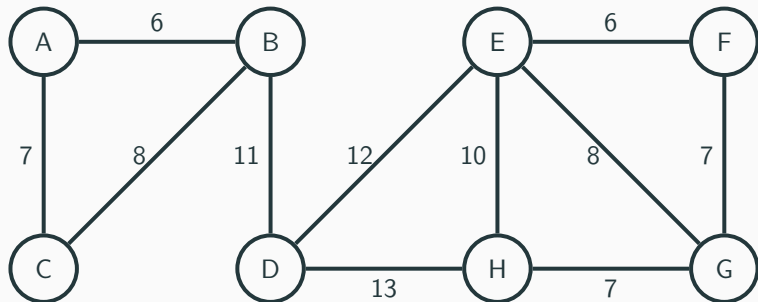
„the fastest“

Borůvka, Kruskal, Prim	$O(m \log(n))$	(deterministisch)
Chazelle	$O(m \log(\beta(m, n)))$	(deterministisch)
<i>MST</i>	$O(m + n)$	(randomisiert)

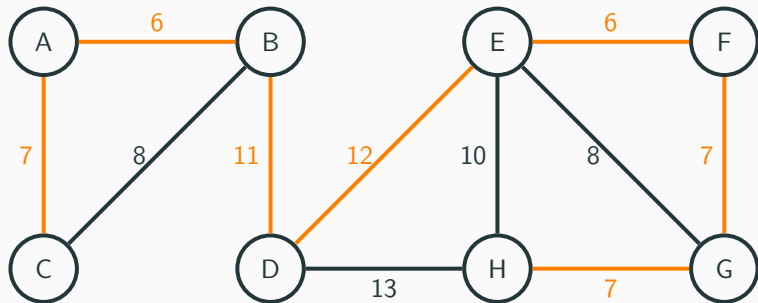
„For many applications, a randomized algorithm is the simplest algorithm available, or the fastest, or both.“[1]

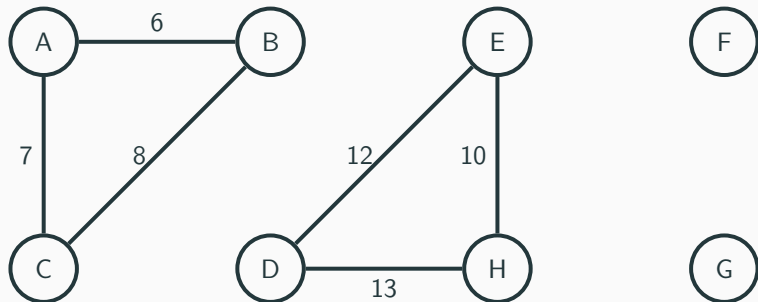
Was wollen wir erreichen?

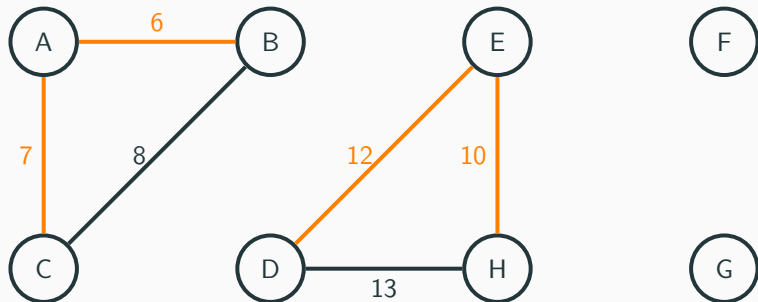
MST



MST



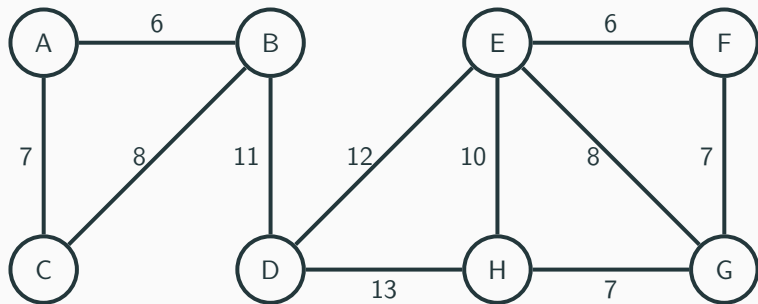




F-leicht/-schwer

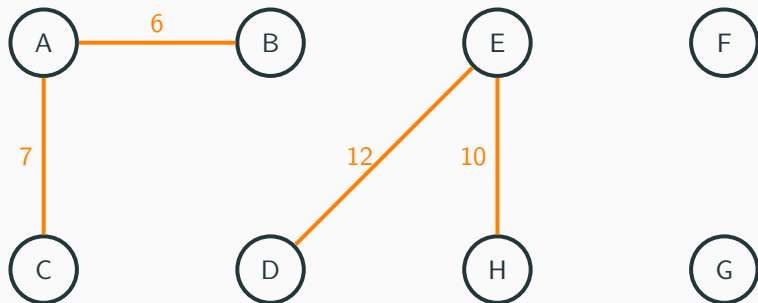
Teaser: F -schwer

Sei G :



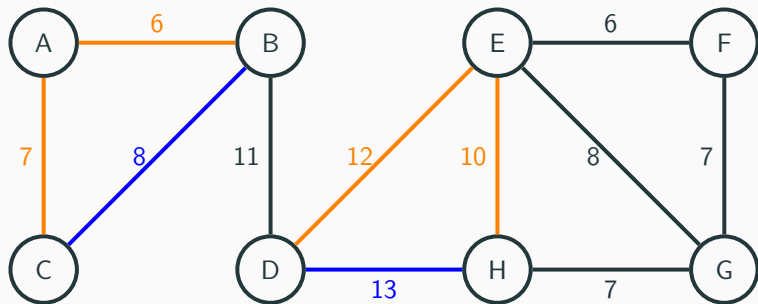
Teaser: F -schwer

Sei F :



Teaser: F -schwer

Dann ist etwas an diesen Kanten besonders.



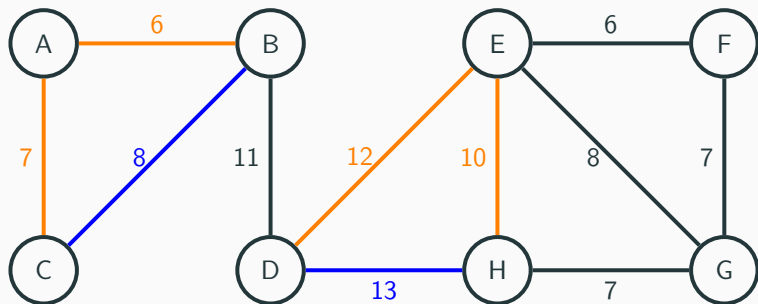
Sei $e = \{u, v\}$, P_e in F , w von G

$$w_F(e) = \begin{cases} \infty & , u \text{ und } v \text{ in verschiedenen Komponenten} \\ \max\{w(P_e(e))\} & , \text{sonst} \end{cases}$$

F-schwer: $w(e) > w_F(e)$

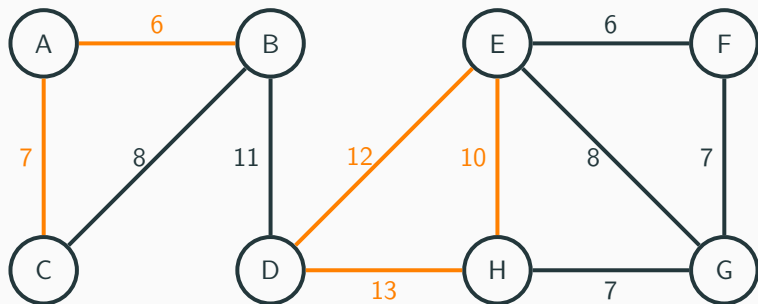
F-leicht: $w(e) \leq w_F(e)$

F-schwere Kanten im MSF?



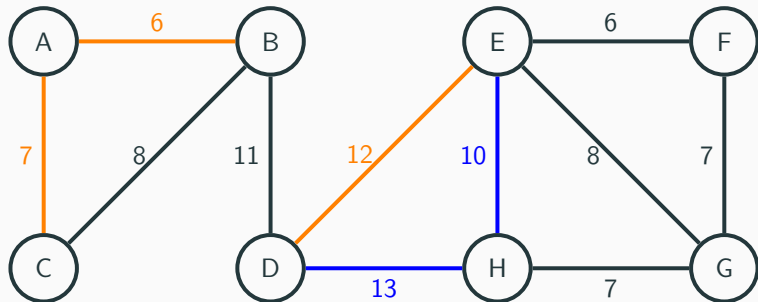
F-schwere Kanten im MSF?

Zyklus D,E,H,D



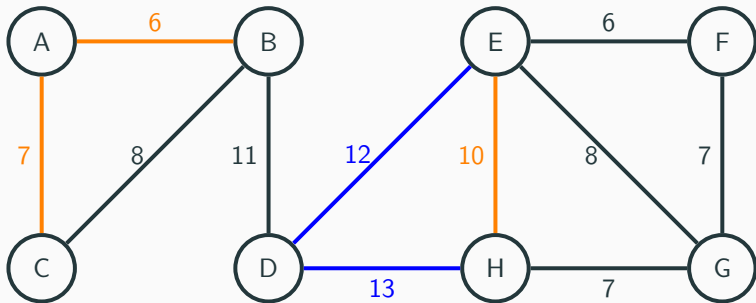
F-schwere Kanten im MSF?

$$w(\{D,H\}) > w(\{E,H\})$$



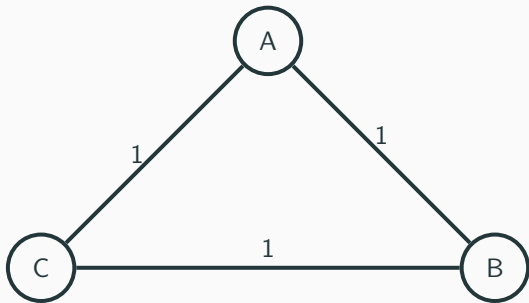
F-schwere Kanten im MSF?

$$w(\{D,H\}) > w(\{D,E\})$$



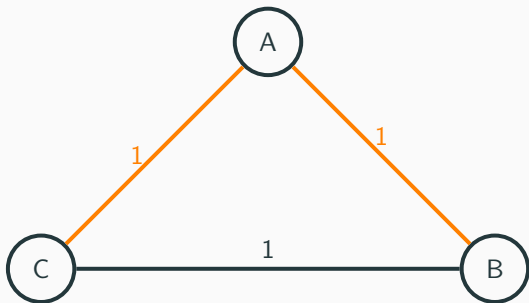
F-leichte Kanten im MSF?

G_{w_1} :



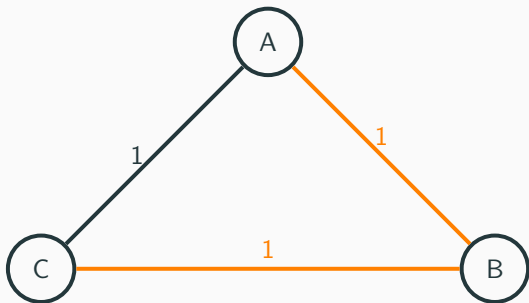
F-leichte Kanten im MSF?

G_{w_1} , MST F :



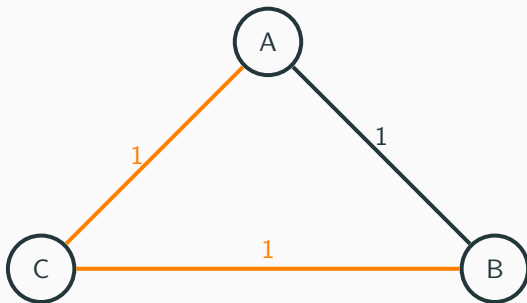
F-leichte Kanten im MSF?

G_{w_1} , MST F :



F-leichte Kanten im MSF?

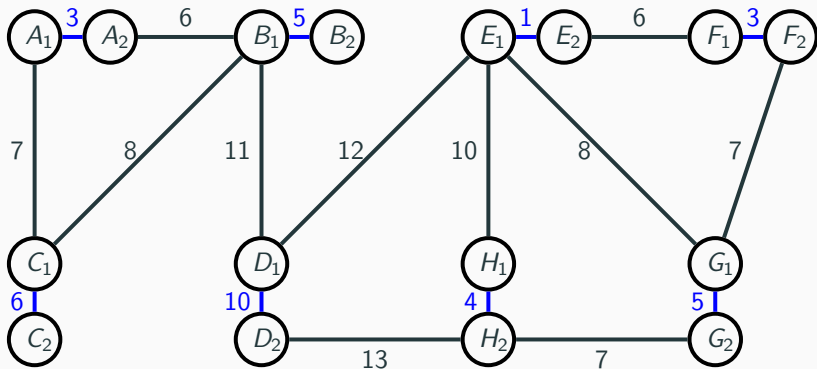
G_{w_1} , MST F :



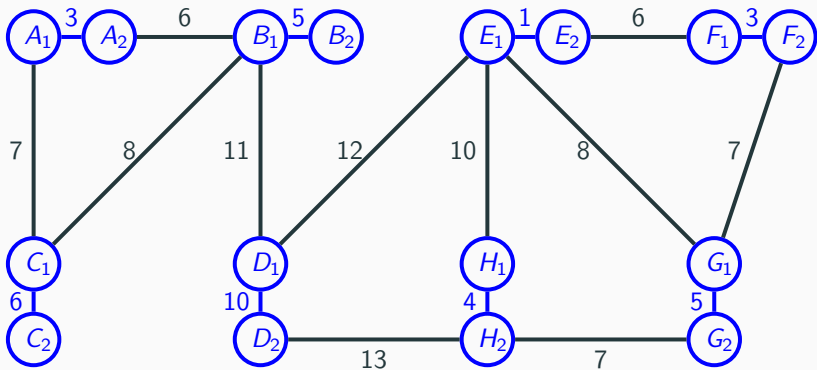
Borůvka Phasen

1. Kontraktierende Kanten markieren
2. Verbundene Komponenten bestimmen
3. Verbundene Komponenten durch einzelne Knoten ersetzen
4. Selbstschleifen entfernen

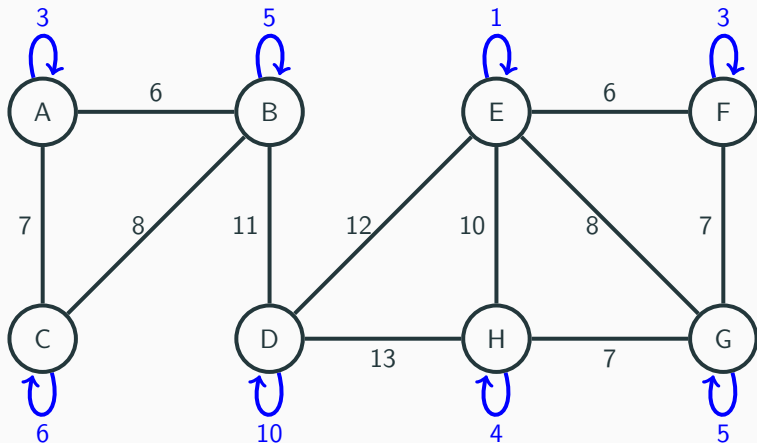
1. Kontraktierende Kanten markieren



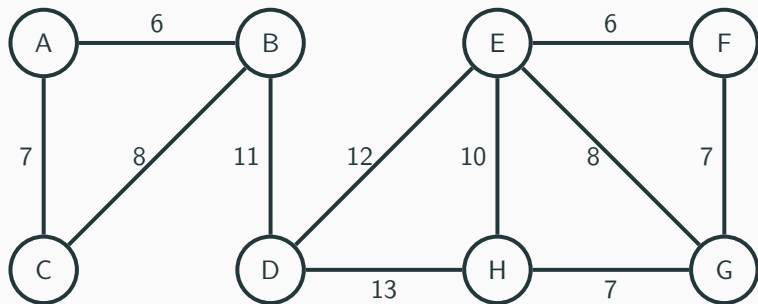
2. Verbundene Komponenten bestimmen



3. Verbundene Komponenten durch einzelne Knoten ersetzen



4. Selbstschleifen entfernen



- Minimale Komponente: zwei Knoten, eine Kante

- Minimale Komponente: zwei Knoten, eine Kante
- Maximale Anzahl minimaler Komponenten: $n/2$

- Minimale Komponente: zwei Knoten, eine Kante
- Maximale Anzahl minimaler Komponenten: $n/2$
- Maximale Anzahl an Knoten nach Borůvka-Phase: $n/2$

Randomisierte Stichproben

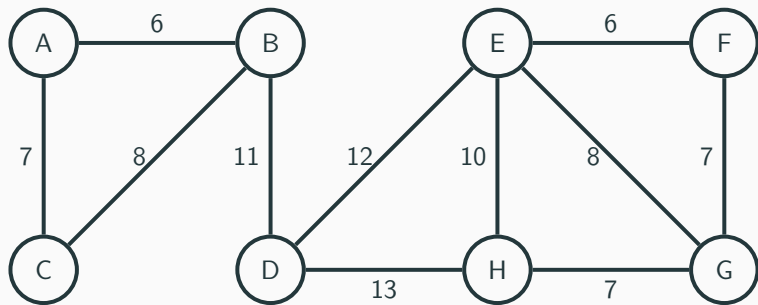
Wirf eine Münze!



Quelle: <https://melbournechapter.net/explore/coin-flip-clipart/>

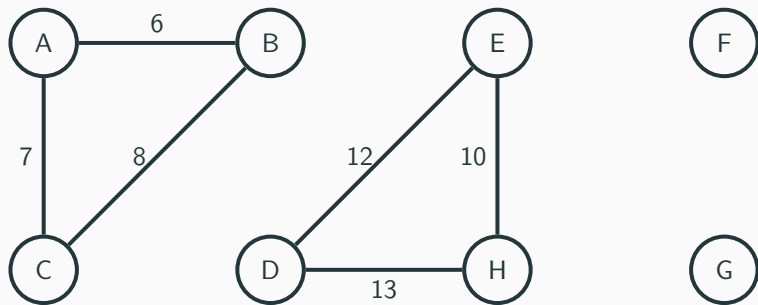
Kanten „würfeln“

G_1 :



Kanten „würfeln“

$G_1(p = 0,5)$:



Verschlechtern wir den MSF?

Theorem

Für den MSF F von $G(p)$, $p \in (0, 1]$ gibt es $\frac{n}{p}$ F -leichte Kanten in G

Beweisidee:

Verschlechtern wir den MSF?

Theorem

Für den MSF F von $G(p)$, $p \in (0, 1]$ gibt es $\frac{n}{p}$ F -leichte Kanten in G

Beweisidee:

- Seien die Kanten von G aufsteigend sortiert

$$e_1, \dots, e_{m_G}, \quad w(e_1) \leq \dots \leq w(e_{m_G})$$

Verschlechtern wir den MSF?

Theorem

Für den MSF F von $G(p)$, $p \in (0, 1]$ gibt es $\frac{n}{p}$ F -leichte Kanten in G

Beweisidee:

- Seien die Kanten von G aufsteigend sortiert
- Sei $F = (V_G, \emptyset)$

Verschlechtern wir den MSF?

Theorem

Für den MSF F von $G(p)$, $p \in (0, 1]$ gibt es $\frac{n}{p}$ F -leichte Kanten in G

Beweisidee:

- Seien die Kanten von G aufsteigend sortiert
- Sei $F = (V_G, \emptyset)$
- Konstruiere $G(p)$ nach der Kantenreihenfolge

Verschlechtern wir den MSF?

Theorem

Für den MSF F von $G(p)$, $p \in (0, 1]$ gibt es $\frac{n}{p}$ F -leichte Kanten in G

Beweisidee:

- Seien die Kanten von G aufsteigend sortiert
- Sei $F = (V_G, \emptyset)$
- Konstruiere $G(p)$ nach der Kantenreihenfolge
- Ist die betrachtete Kante F -leicht wird sie in F aufgenommen

- Wann wird die nächste Kante hinzugenommen?

- Wann wird die nächste F -leichte Kante hinzugenommen?

- Wann wird die nächste F -leichte Kante hinzugenommen?

k -te Phase $\stackrel{\text{def}}{=} \quad$ Zufallsexperimente ab $k - 1$ Kanten in F ,
bis k Kanten in F

- Wann wird die nächste F -leichte Kante hinzugenommen?
- Wie oft „würfeln“?

k -te Phase $\stackrel{\text{def}}{=}$ Zufallsexperimente ab $k - 1$ Kanten in F ,
bis k Kanten in F

- Wann wird die nächste F -leichte Kante hinzugenommen?
- Wie oft „würfeln“? - $1/p$

k -te Phase $\stackrel{\text{def}}{=}$ Zufallsexperimente ab $k - 1$ Kanten in F ,
bis k Kanten in F

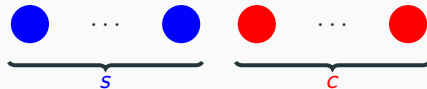
- Wann wird die nächste F -leichte Kante hinzugenommen?
- Wie oft „würfeln“? - $1/p$
- Anzahl an Phasen insgesamt: $s \leq n - 1, s \in \mathbb{N}$

k -te Phase $\stackrel{\text{def}}{=}$ Zufallsexperimente ab $k - 1$ Kanten in F ,
bis k Kanten in F

- Differenz der Kanten $c = (n - 1) - s < n - s$

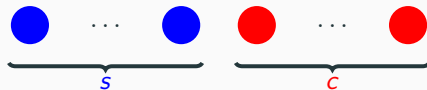
Beweis: Stochastische Dominanz

- Differenz der Kanten $c = (n - 1) - s < n - s$



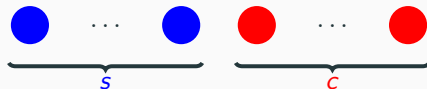
Beweis: Stochastische Dominanz

- Differenz der Kanten $c = (n - 1) - s < n - s$
- weitere c -Phasen \Rightarrow erwartet mehr F -leichte Kanten in G



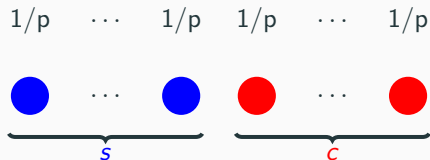
Beweis: Stochastische Dominanz

- Differenz der Kanten $c = (n - 1) - s < n - s$
- weitere c -Phasen \Rightarrow erwartet mehr F -leichte Kanten in G
- $s + c = n - 1$ Phasen, bzw. $n - 1$ Erfolge



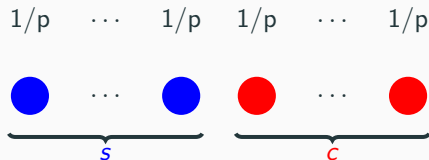
Beweis: Stochastische Dominanz

- Differenz der Kanten $c = (n - 1) - s < n - s$
- weitere c -Phasen \Rightarrow erwartet mehr F -leichte Kanten in G
- $s + c = n - 1$ Phasen, bzw. $n - 1$ Erfolge

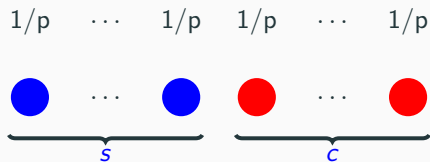


Beweis: Stochastische Dominanz

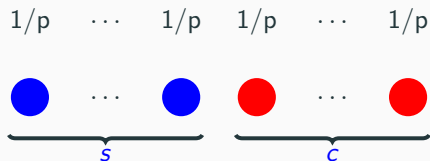
- Differenz der Kanten $c = (n - 1) - s < n - s$
- weitere c -Phasen \Rightarrow erwartet mehr F -leichte Kanten in G
- $s + c = n - 1$ Phasen, bzw. $n - 1$ Erfolge
- Erfolgswahrscheinlichkeit: $p \Rightarrow$ negative Binomialverteilung



Beweis: Stochastische Dominanz



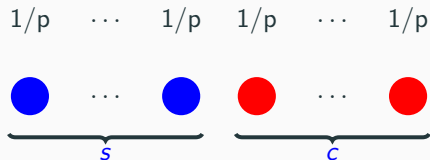
Beweis: Stochastische Dominanz



$X_{np} \stackrel{\text{def}}{=} \text{negative Binomialverteilung, Parameter } n - 1, p$

$X_{sp} \stackrel{\text{def}}{=} \text{negative Binomialverteilung, Parameter } s, p$

Beweis: Stochastische Dominanz



$X_{np} \stackrel{\text{def}}{=} \text{negative Binomialverteilung, Parameter } n - 1, p$

$X_{sp} \stackrel{\text{def}}{=} \text{negative Binomialverteilung, Parameter } s, p$

X_{np} dominiert X_{sp} , mit:

$$n/p > (n - 1)/p = E[X_{np}] \geq E[X_{sp}] = s/p$$

Der MST-Algorithmus

MST

Data: Graph G

Result: Approximation eines MST/ MSF in G

MST

Data: Graph G

Result: Approximation eines MST/ MSF in G

3 Borůvka-Phasen

1: $G_1, C \leftarrow$ **Wenn** G leer oder Borůvka-Phasen terminieren:
 return $F = C$

MST

Data: Graph G

Result: Approximation eines MST/ MSF in G

3 Borůvka-Phasen

1: $G_1, C \leftarrow$ **Wenn** G leer oder Borůvka-Phasen terminieren:

return $F = C$

2: $G_2 \leftarrow G_1(p = 0,5)$

MST

Data: Graph G

Result: Approximation eines MST/ MSF in G

3 Borůvka-Phasen

- 1: $G_1, C \leftarrow$ **Wenn** G leer oder Borůvka-Phasen terminieren:
 return $F = C$
- 2: $G_2 \leftarrow G_1(p = 0,5)$
- 3: $F_2 \leftarrow MST(G_2)$

3 Borůvka-Phasen

$G_1, C \leftarrow$ **Wenn** G leer
oder Borůvka-Phasen terminieren:
return $F = C$

$G_2 \leftarrow G_1(p = 0, 5)$

$F_2 \leftarrow MST(G_2)$

$G_3 \leftarrow (V_{G_1}, E_{G_1} - E_{F_2-heavy})$

$F_3 \leftarrow MST(G_3)$

return $F = C \cup F_3$

3 Borůvka-Phasen

$O(n + m)$ $G_1, C \leftarrow$ **Wenn** G leer
oder Borůvka-Phasen terminieren:
return $F = C$

$O(n + m)$ $G_2 \leftarrow G_1(p = 0, 5)$

$F_2 \leftarrow MST(G_2)$

$O(n + m)$ $G_3 \leftarrow (V_{G_1}, E_{G_1} - E_{F_2-heavy})$

$F_3 \leftarrow MST(G_3)$

$O(n + m)$ **return** $F = C \cup F_3$

3 Borůvka-Phasen

$O(n + m)$ $G_1, C \leftarrow$ **Wenn** G leer
oder Borůvka-Phasen terminieren:
return $F = C$

$O(n + m)$ $G_2 \leftarrow G_1(p = 0, 5)$

? $F_2 \leftarrow MST(G_2)$

$O(n + m)$ $G_3 \leftarrow (V_{G_1}, E_{G_1} - E_{F_2-heavy})$

? $F_3 \leftarrow MST(G_3)$

$O(n + m)$ **return** $F = C \cup F_3$

$$T(m, n) \leq ? + ? + c(m + n)$$

, mit $c \in \mathbb{N}$ konstant

$$T(m, n) \leq ? + ? + c(m + n)$$

, mit $c \in \mathbb{N}$ konstant

$$G_1 \quad n_{G_1} = n_G/8, m_{G_1} = m_G$$

$$T(m, n) \leq ? + ? + c(m + n)$$

, mit $c \in \mathbb{N}$ konstant

G_1	$n_{G_1} = n_G/8, m_{G_1} = m_G$
G_2	$n_{G_2} = n_G/8, m_{G_2} = m_G/2$

$$T(m, n) \leq T(n/8, m/2) + ? + c(m + n)$$

, mit $c \in \mathbb{N}$ konstant

G_1	$n_{G_1} = n_G/8, m_{G_1} = m_G$
G_2	$n_{G_2} = n_G/8, m_{G_2} = m_G/2$
G_3	$n_{G_3} = \frac{n_G/8}{0,5}, m_{G_3} = m_G/2$

$$T(m, n) \leq T(n/8, m/2) + T(n/8, n/4) + c(m + n)$$

, mit $c \in \mathbb{N}$ konstant

G_1	$n_{G_1} = n_G/8, m_{G_1} = m_G$
G_2	$n_{G_2} = n_G/8, m_{G_2} = m_G/2$
G_3	$n_{G_3} = n_G/4, m_{G_3} = m_G/2$

$$\begin{aligned}T(m, n) &\leq T(n/8, m/2) + T(n/8, n/4) + c(m + n) \\&\leq (T(n/8^2, m/2^2) + T(n/8^2, \frac{m/2}{4}) + c(n/8 + m/2)) \\&\quad + (T(n/8^2, \frac{n/4}{2}) + T(n/8^2, n/4^2) + c(n/8 + n/4)) \\&\quad + c(n + m)\end{aligned}$$

, mit $c \in \mathbb{N}$ konstant

$$\begin{aligned}T(m, n) &\leq T(n/8, m/2) + T(n/8, n/4) + c(m + n) \\&\leq (T(n/8^2, m/2^2) + T(n/8^2, \frac{m/2}{4}) \\&\quad + T(n/8^2, \frac{n/4}{2}) + T(n/8^2, n/4^2) \\&\quad + c(n/8 + n/4) + c(n/8 + m/2)) + c(n + m)\end{aligned}$$

, mit $c \in \mathbb{N}$ konstant

$$\begin{aligned}T(m, n) &\leq T(n/8, m/2) + T(n/8, n/4) + c(m + n) \\&\leq (T(n/8^2, m/2^2) + T(n/8^2, \frac{m/2}{4}) \\&\quad + T(n/8^2, \frac{n/4}{2}) + T(n/8^2, n/4^2) \\&\quad + c(n/2 + m/2)) + c(n + m)\end{aligned}$$

, mit $c \in \mathbb{N}$ konstant

$$\begin{aligned}T(m, n) &\leq T(n/8, m/2) + T(n/8, n/4) + c(m + n) \\&\leq (T(n/8^2, m/2^2) + T(n/8^2, \frac{m/2}{4}) \\&\quad + T(n/8^2, \frac{n/4}{2}) + T(n/8^2, n/4^2) \\&\quad + c(n/2 + m/2)) + c(n + m) \\&\leq c(n + m) \cdot \sum_{i=0}^{\infty} (1/2)^i\end{aligned}$$

, mit $c \in \mathbb{N}$ konstant

$$\begin{aligned}T(m, n) &\leq T(n/8, m/2) + T(n/8, n/4) + c(m + n) \\&\leq (T(n/8^2, m/2^2) + T(n/8^2, \frac{m/2}{4}) \\&\quad + T(n/8^2, \frac{n/4}{2}) + T(n/8^2, n/4^2) \\&\quad + c(n/2 + m/2)) + c(n + m) \\&\leq c(n + m) \cdot \sum_{i=0}^{\infty} (1/2)^i \\&= 2 \cdot c(n + m)\end{aligned}$$

, mit $c \in \mathbb{N}$ konstant

: Motwani, R., Raghavan, P. : *Randomized Algorithms*. Cambridge : Cambridge University Press 1995, Kapitel 10.3.