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)
MST	$O(m+n)$	(randomisiert)

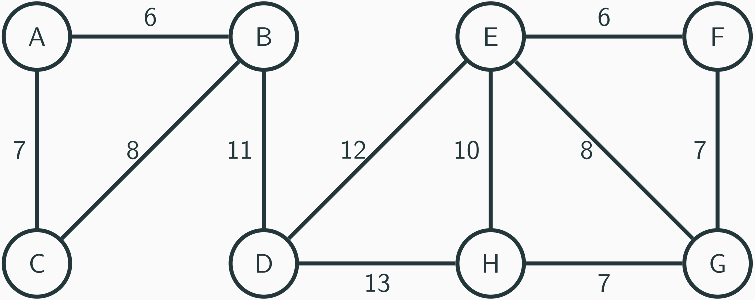
„the fastest“

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

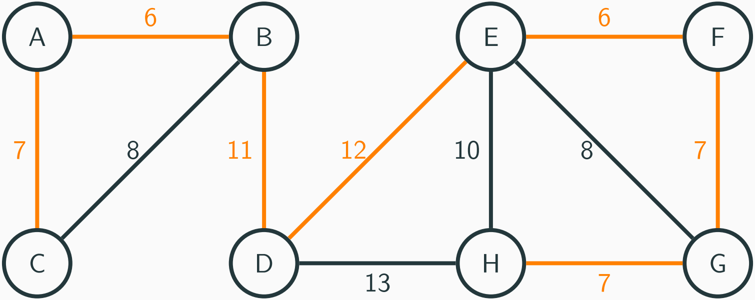
„For many applications, a randomized algorithm is the simplest algorithm available, or the fastest, or both.“[1]

Was wollen wir erreichen?

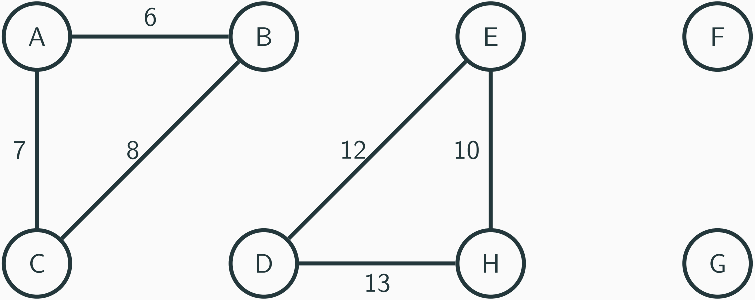
MST



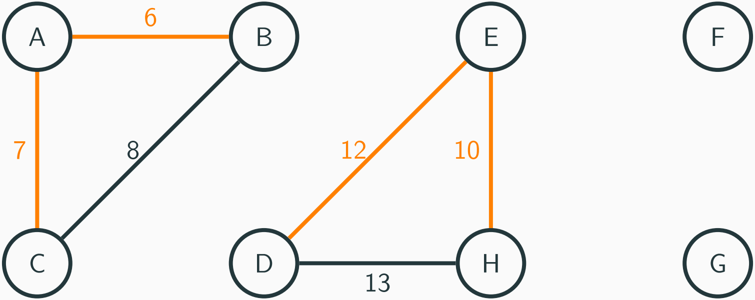
MST



MSF



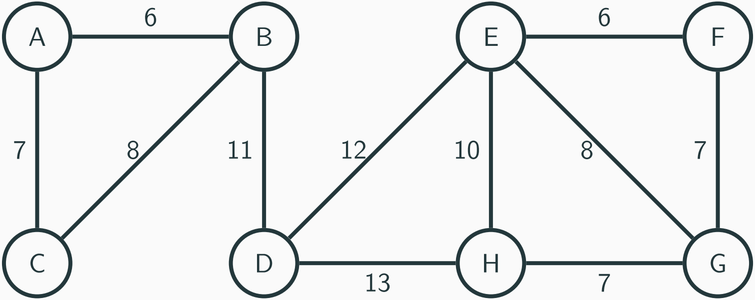
MSF



F-leicht/-schwer

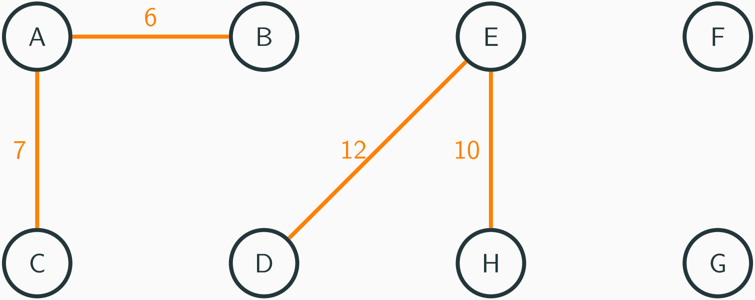
Teaser: F-schwer

Sei G:



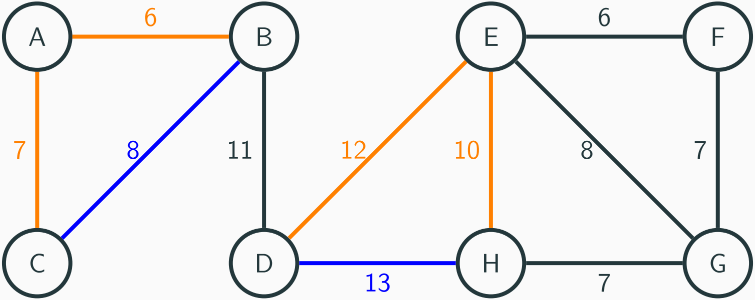
Teaser: F-schwer

Sei F:



Teaser: F-schwer

Dann ist etwas an diesen Kanten besonders.



F-leicht/-schwer

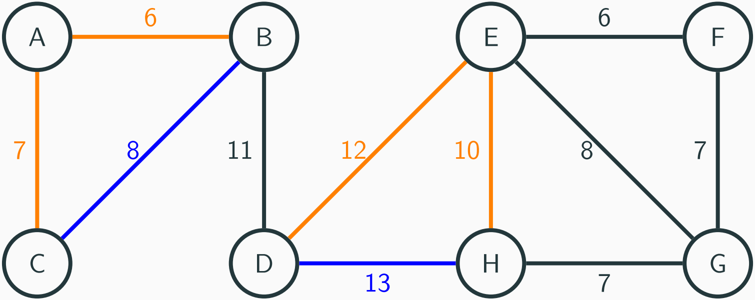
Sei $e = \{u, v\}$, P_e in F, w von G

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

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

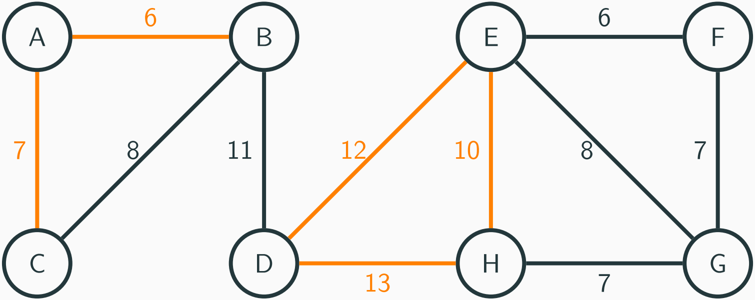
F-leicht: $w(e) \leq w_F(e)$

F-schwere Kanten im MSF?



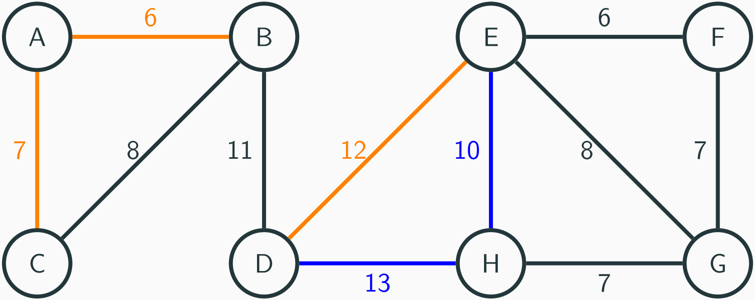
F-schwere Kanten im MSF?

Zyklus D,E,H,D ↯



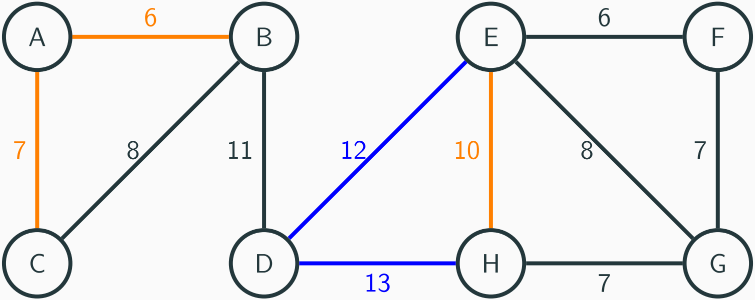
F-schwere Kanten im MSF?

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



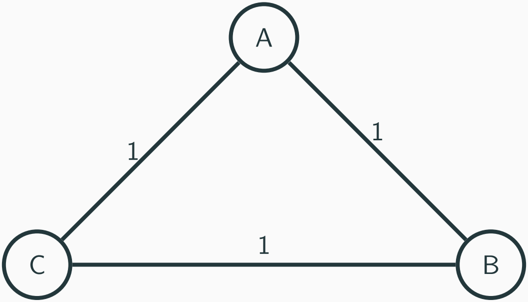
F-schwere Kanten im MSF?

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



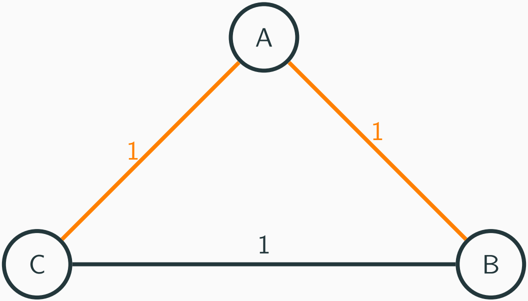
F-leichte Kanten im MSF?

G_{w_1}:



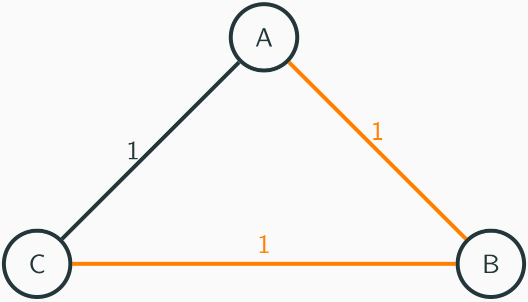
F-leichte Kanten im MSF?

G_{w_1}, MST F:



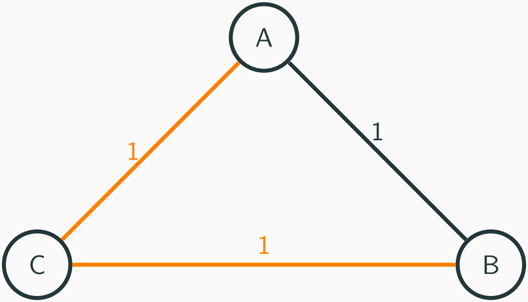
F-leichte Kanten im MSF?

G_{w_1}, MST F:



F-leichte Kanten im MSF?

G_{w_1}, MST F:



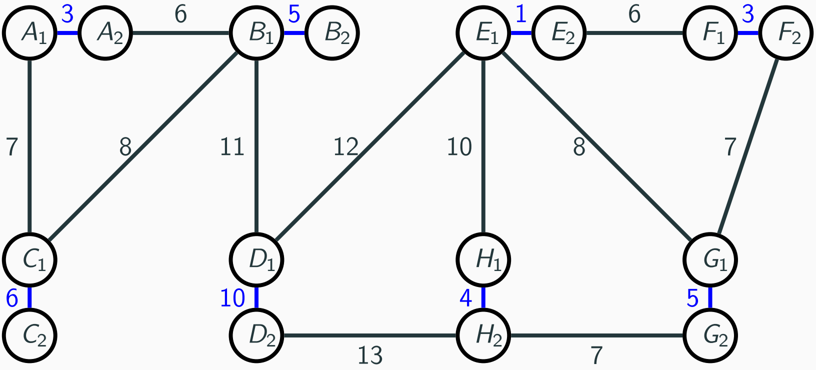
Borůvka Phasen

Ablauf

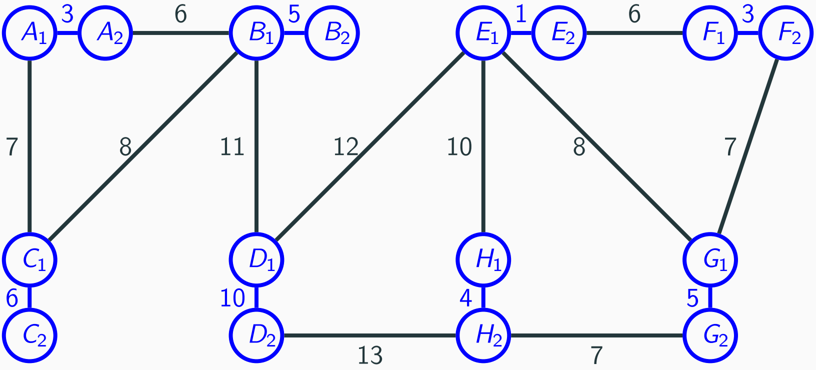
1. Kontraktierende Kanten markieren
2. Verbundene Komponenten bestimmen
3. Verbundene Komponenten durch einzelne Knoten ersetzen
4. Selbstschleifen entfernen

Ablauf

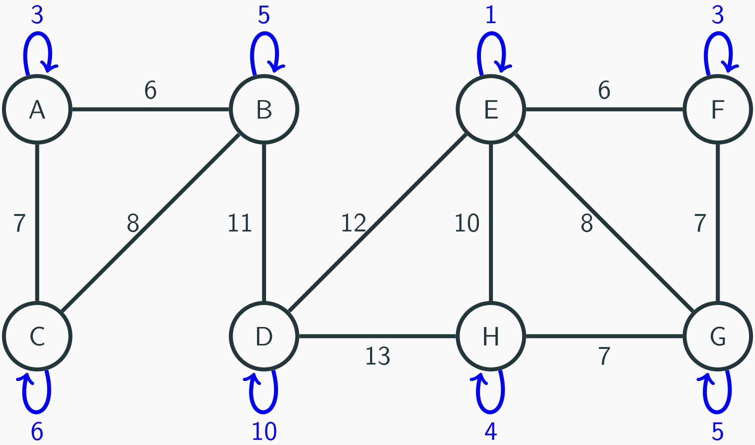
1. Kontraktierende Kanten markieren



2. Verbundene Komponenten bestimmen

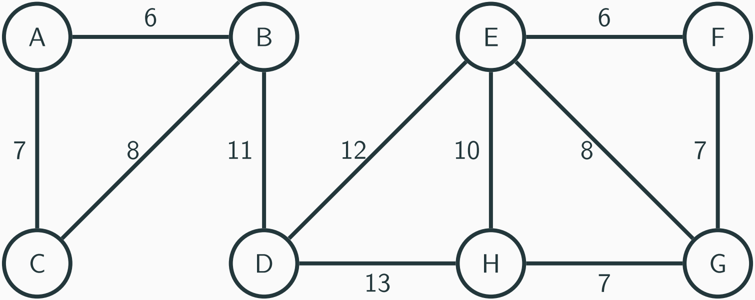


3. Verbundene Komponenten durch einzelne Knoten ersetzen



Ablauf

4. Selbstschleifen entfernen



Reduktion der Knoten

- Minimale Komponente: zwei Knoten, eine Kante

Reduktion der Knoten

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

Reduktion der Knoten

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

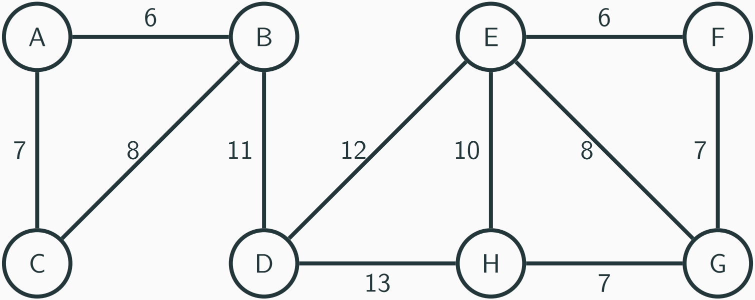
Randomisierte Stichproben

Wirf eine Münze!

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

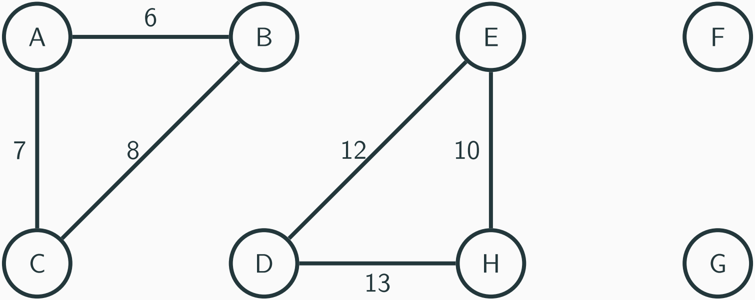
Kanten „würfeln“

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

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

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

Beweis: Phasen

- Wann wird die nächste Kante hinzugenommen?

Beweis: Phasen

- Wann wird die nächste F-leichte Kante hinzugenommen?

Beweis: Phasen

- Wann wird die nächste F-leichte Kante hinzugenommen?

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

Beweis: Phasen

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

$$k\text{-te Phase} \overset{def}{=} \quad \text{Zufallsexperimente} \quad \begin{array}{l} \text{ab } k-1 \text{ Kanten in } F, \\ \text{bis } k \text{ Kanten in } F \end{array}$$

Beweis: Phasen

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

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

Beweis: Phasen

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

Beweis: Stochastische Dominanz

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

Beweis: Stochastische Dominanz

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

Beweis: Stochastische Dominanz

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

Beweis: Stochastische Dominanz

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

Beweis: Stochastische Dominanz

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

$X_{np} \overset{def}{=}$ negative Binomialverteilung, Parameter n, p
$X_{sp} \overset{def}{=}$ negative Binomialverteilung, Parameter s, p
X_{np} dominiert X_{sp}, mit:

$$n/p = E[X_{np}] \geq E[X_{sp}] = s/p$$

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

1: $G_1, C \leftarrow$ 3 Borůvka-Phasen

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

1: $G_1, C \leftarrow$ 3 Borůvka-Phasen
Wenn G leer oder Borůvka-Phasen terminieren:
return $F = C$

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

3: $F_2 \leftarrow MST(G_2)$

MST

Data: Graph G

Result: Approximation eines MST/ MSF in G

3 Borůvka-Phasen

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

return $F = C$

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

$F_2 \leftarrow MST(G_2)$

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

MST

Data: Graph G

Result: Approximation eines MST/ MSF in G

$G_1, C \leftarrow$ 3 Borůvka-Phasen
Wenn G leer oder Borůvka-Phasen terminieren:
return $F = C$
$G_2 \leftarrow G_1(p = 0,5)$
$F_2 \leftarrow MST(G_2)$
$G_3 \leftarrow (V_{G_1}, E_{G_1} - E_{F_2-heavy})$
$F_3 \leftarrow MST(G_3)$

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: $F_2 \leftarrow MST(G_2)$

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

5: $F_3 \leftarrow MST(G_3)$

6: **return** $F = C \cup F_3$

Laufzeit: Überblick

$G_1, C \leftarrow$ 3 Borůvka-Phasen
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 $F = C \cup F_3$

Laufzeit: Überblick

$O(n+m)$ $\quad G_1, C \leftarrow$ 3 Borůvka-Phasen

Wenn G leer

oder Borůvka-Phasen terminieren:

return $F = C$

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

$F_2 \leftarrow MST(G_2)$

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

$F_3 \leftarrow MST(G_3)$

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

Laufzeit: Überblick

$O(n+m)$ $G_1, C \leftarrow$ 3 Borůvka-Phasen
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** $F = C \cup F_3$

Laufzeit: Rekursionsgleicung

$T(m, n) \leq {\color{red}?} + {\color{red}?} + c(m + n)$

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

Laufzeit: Rekursionsgleicung

$T(m, n) \leq {\color{red}?} + {\color{red}?} + c(m + n)$

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

G_1	$n_1 = n/8, m_1 = m$

Laufzeit: Rekursionsgleicung

$$T(m, n) \leq {\color{red}?} + {\color{red}?} + c(m + n)$$

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

G_1	$n_1 = n/8, m_1 = m$
G_2	$n_2 = n/8, m_2 = m/2$

Laufzeit: Rekursionsgleicung

$T(m, n) \leq \mathrm{T(n/8,\ m/2)} + {\color{red}?} + c(m+n)$

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

G_1	$n_1 = n/8, m_1 = m$
G_2	$n_2 = n/8, m_2 = m/2$
G_3	$n_3 = \frac{n/8}{0{,}5}, m_3 = m/2$

Laufzeit: Rekursionsgleicung

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

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

G_1	$n_1 = n/8, m_1 = m$
G_2	$n_2 = n/8, m_2 = m/2$
G_3	$n_3 = n/4, m_3 = m/2$

Laufzeit: Rekursionsgleicung

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

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

Laufzeit: Rekursionsgleicung

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

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

Laufzeit: Rekursionsgleicung

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

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

Laufzeit: Rekursionsgleicung

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

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

Laufzeit: Rekursionsgleicung

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

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

Literatur

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