Lecture: Approximation Algorithms

Jannik Matuschke

TUM

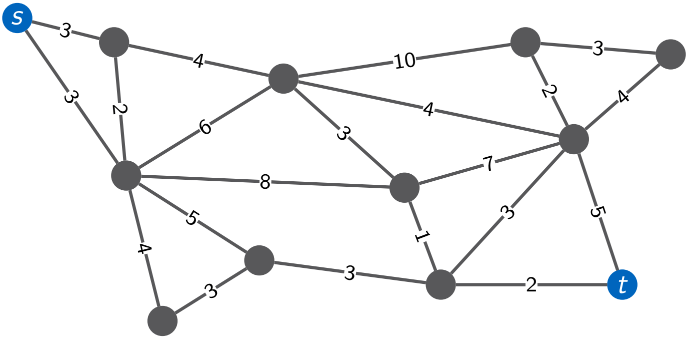
November 21, 2018

The primal-dual method
Example I: The shortest path problem

Shortest Path

Input: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_+$,
start $s \in V$, target $t \in V$

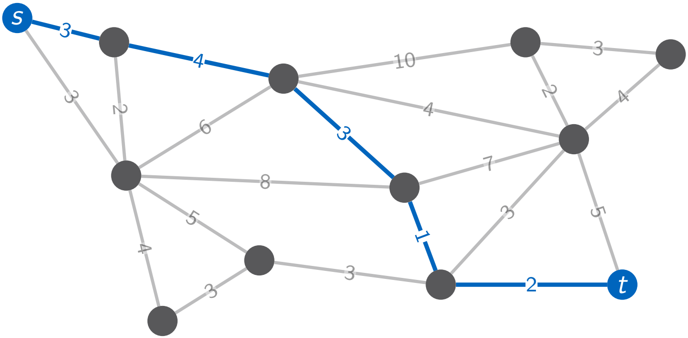
Task: find an s-t-path P minimizing $\sum_{e \in P} w_e$



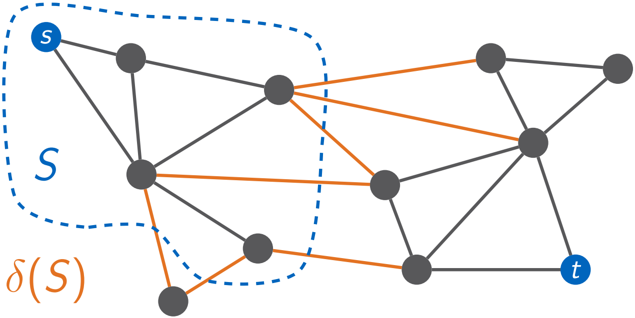
Shortest Path

Input: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_+$,
start $s \in V$, target $t \in V$

Task: find an s-t-path P minimizing $\sum_{e \in P} w_e$

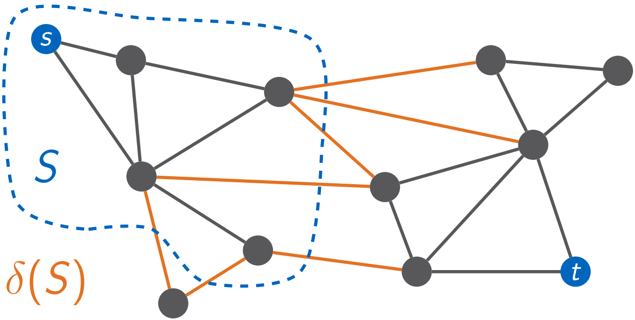


LP relaxation



$$\mathcal{F} := \{S \subseteq V \,:\, s \in S,\, t \notin S\}$$

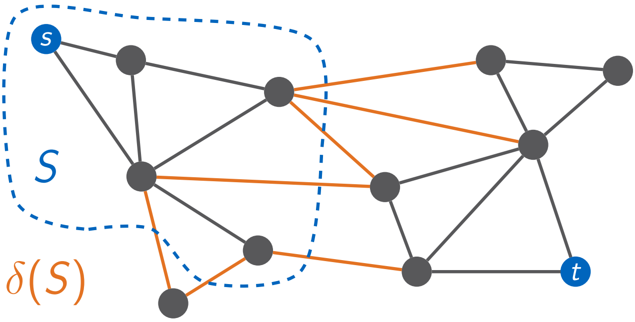
$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e \ \geq \ 1 \qquad \forall\, S \in \mathcal{F} \\ & x_e \ \in \{0,1\} \quad \forall\, e \in E \end{aligned}$$



$$\mathcal{F} := \{S \subseteq V \ : \ s \in S,\ t \notin S\}$$

$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e \ \geq\ 1 \qquad \forall\, S \in \mathcal{F} \\ & x_e \ \in \{0,1\} \qquad \forall\, e \in E \end{aligned}$$

LP relaxation



$$\mathcal{F} := \{S \subseteq V \ : \ s \in S,\ t \notin S\}$$

$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e \geq 1 \quad \forall\, S \in \mathcal{F} \\ & x_e \geq 0 \quad \forall\, e \in E \end{aligned}$$

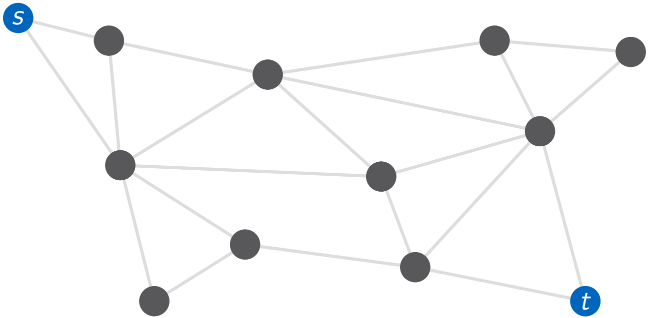
Primal-dual algorithm

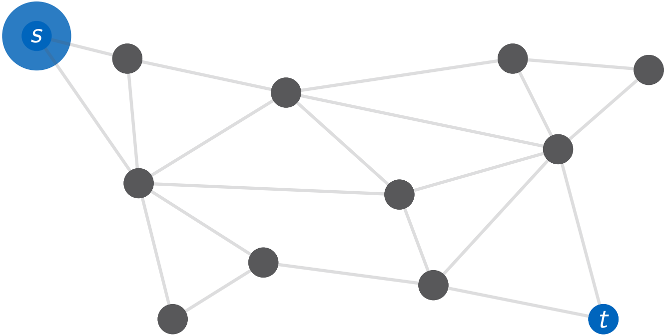
Algorithm

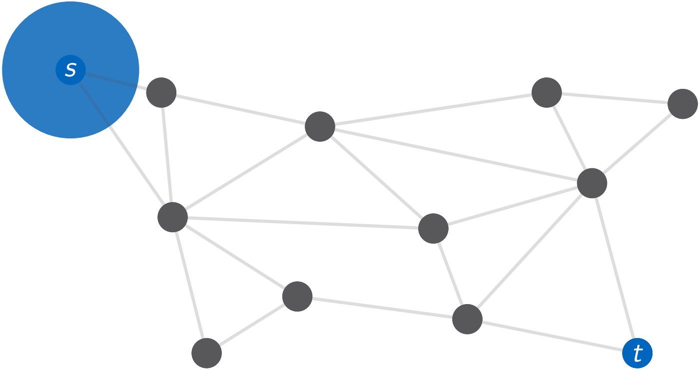
1. $T := \emptyset$, $y := 0$
2. while T does not contain s-t-path
   - Let C be the set of vertices connected to s in (V, T).
   - Increase y_C until there is an $e \in \delta(C)$ with $\sum_{e \in \mathcal{F}_e} y_S = w_e$.
   - $T := T \cup \{e\}$ (increase can be 0)
3. Let P be an s-t-path in T.
4. return P

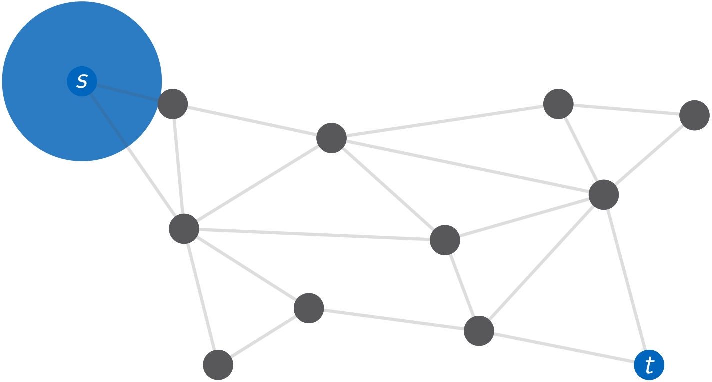
Theorem 9.1

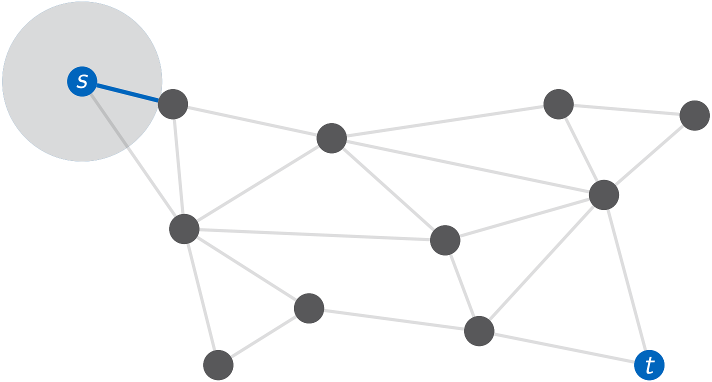
Primal-dual is a 1-approximation algorithm for SHORTEST PATH.

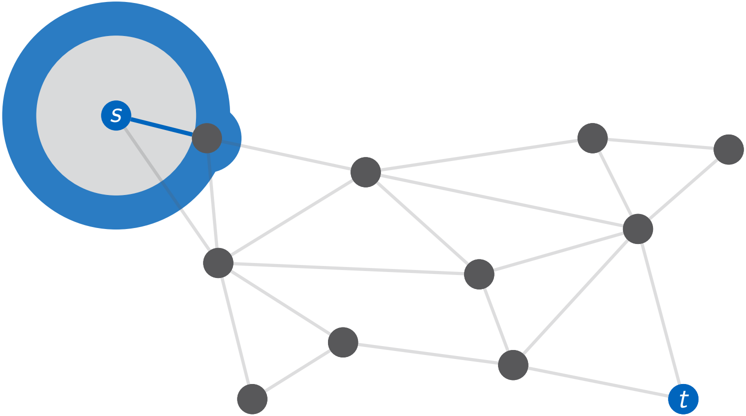

s
t

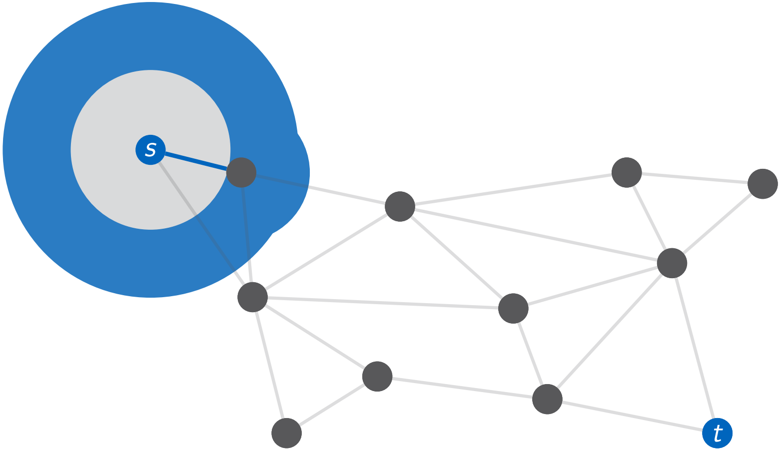

s
t

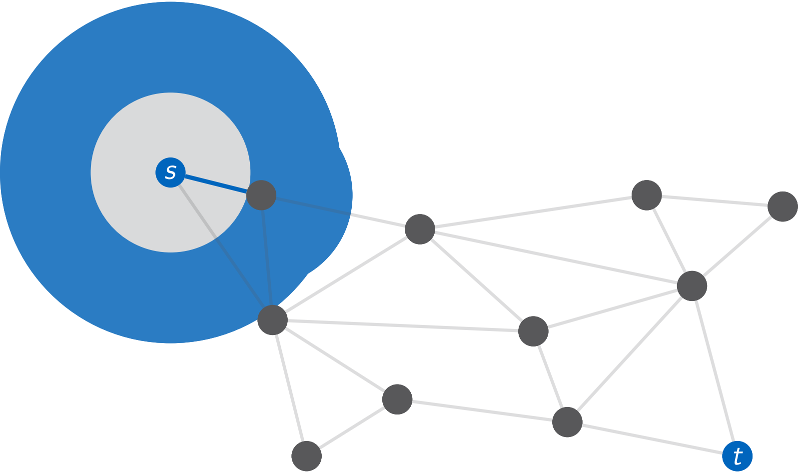

s
t

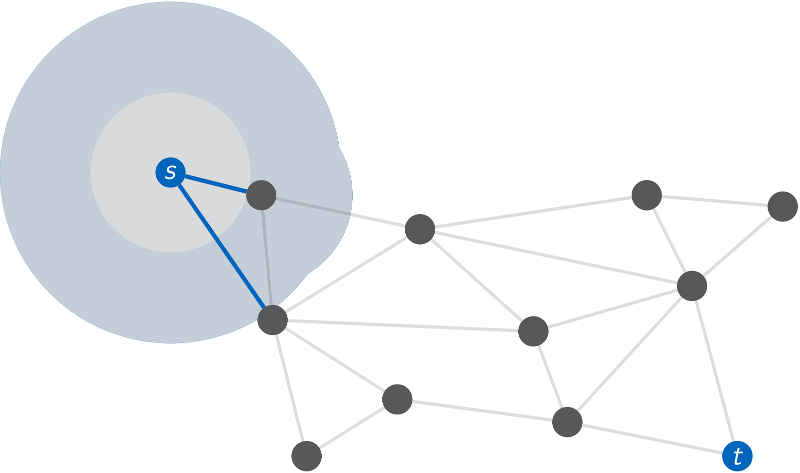

s
t

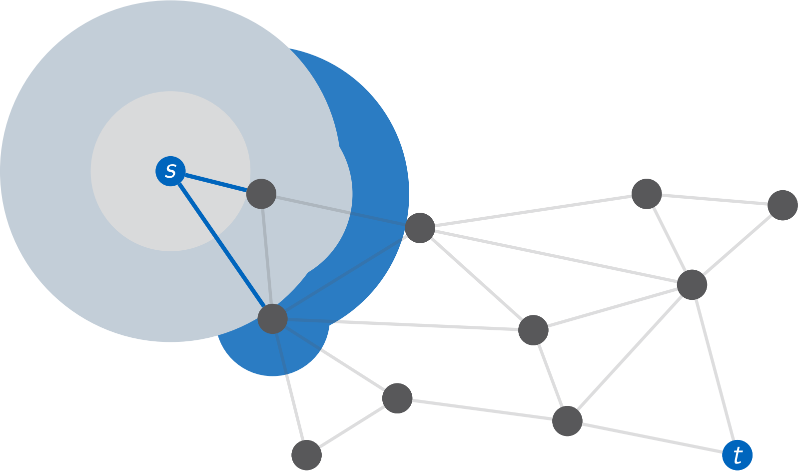

s
t

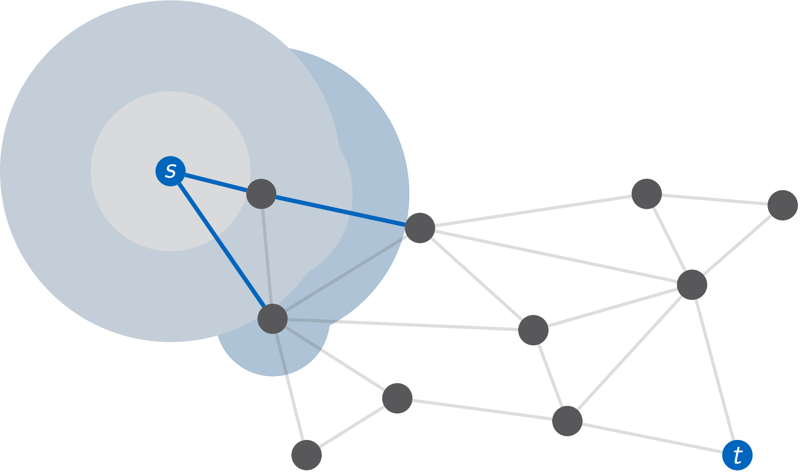

s
t

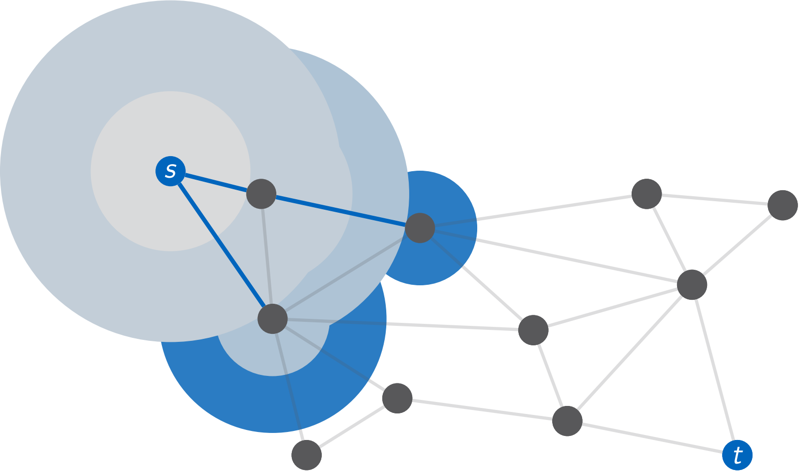

s
t

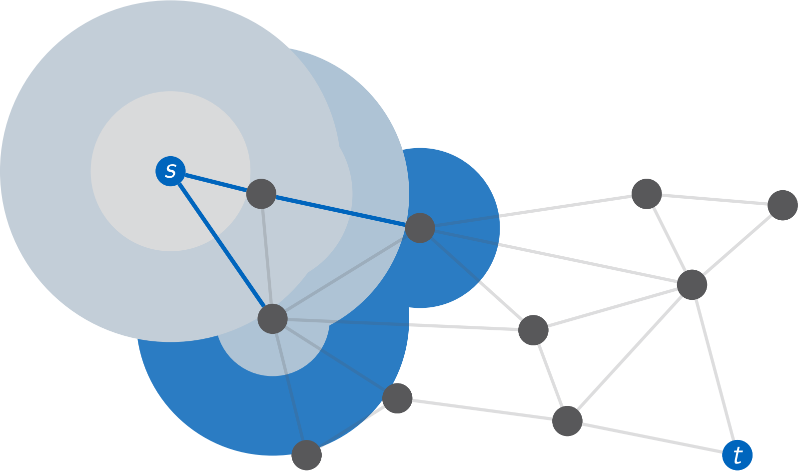

s
t

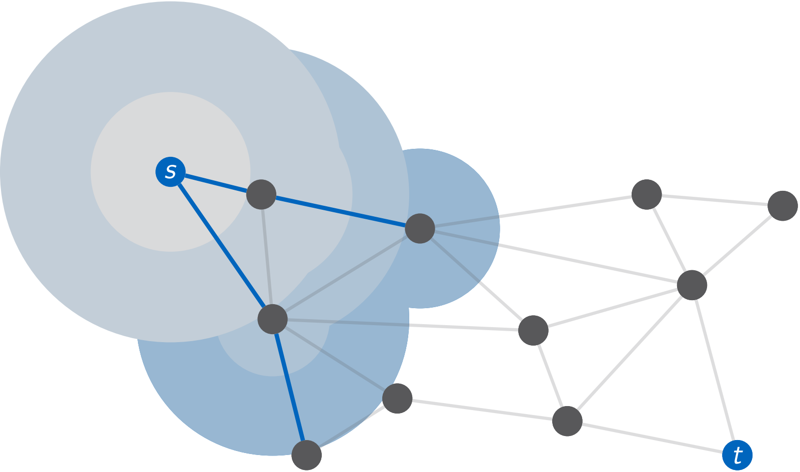

s
t

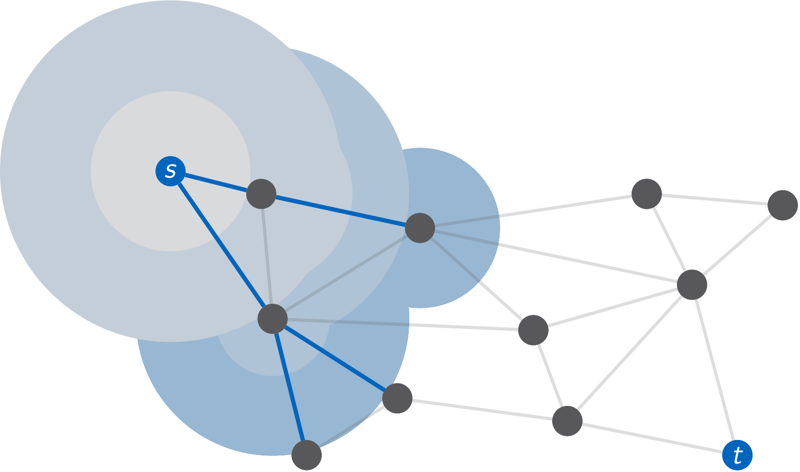

s
t

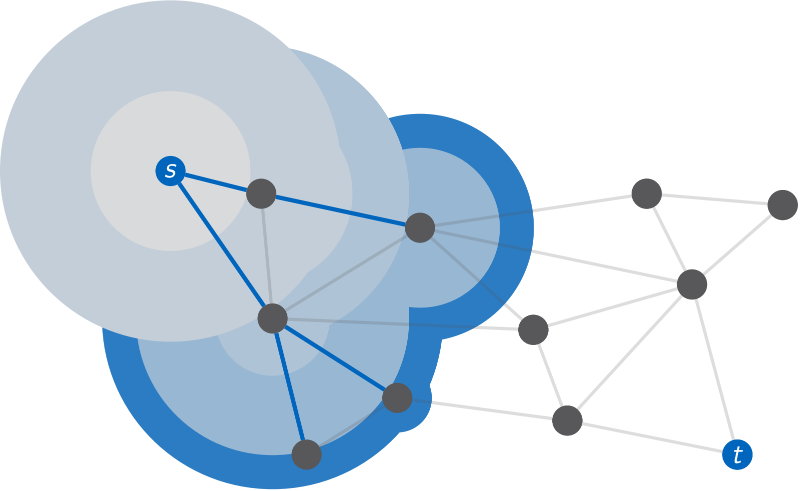

s
t


s
t

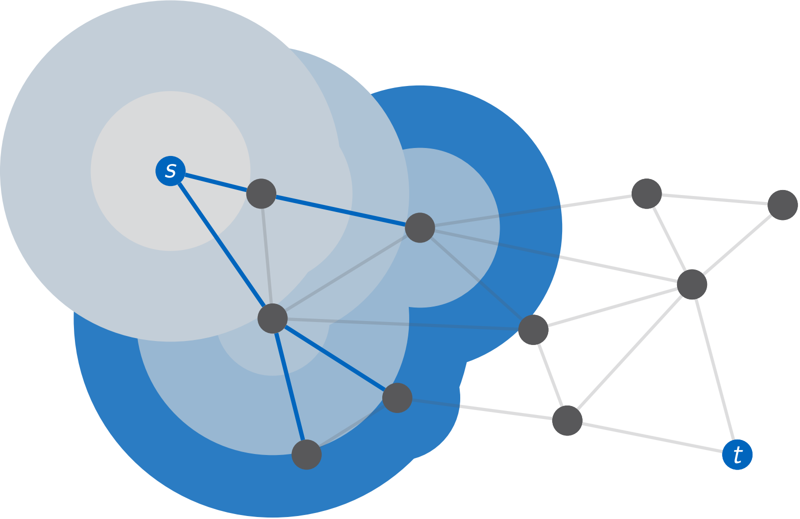

s
t

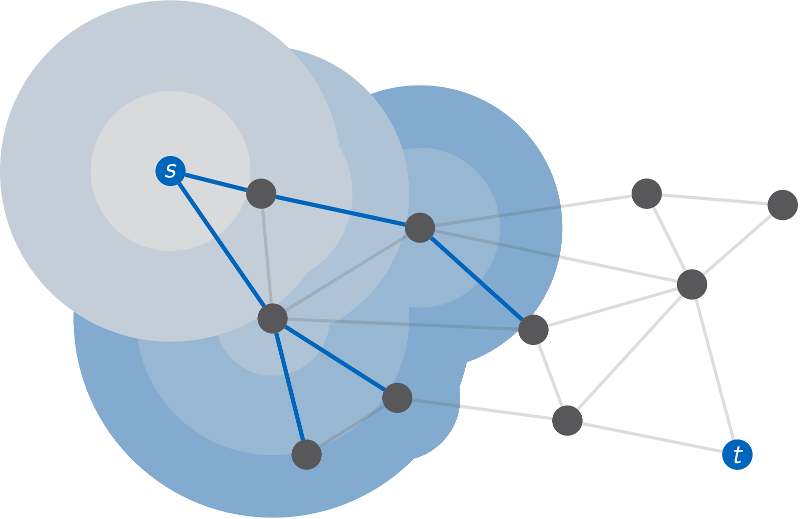

s
t

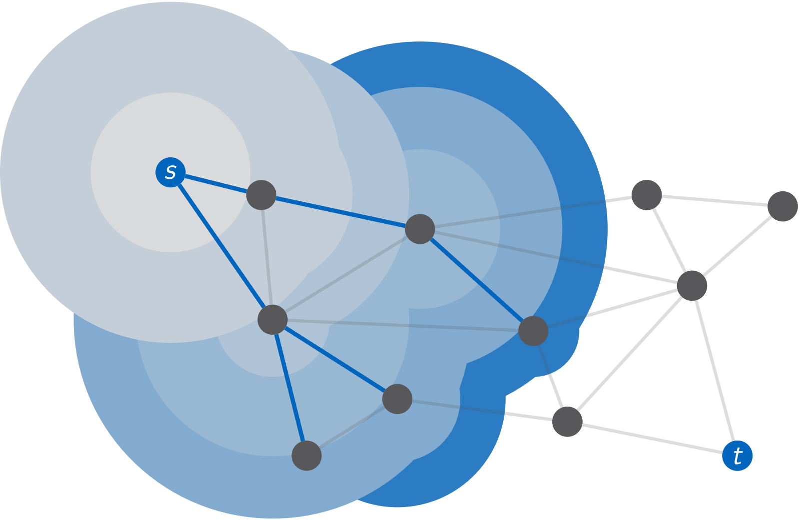

s
t



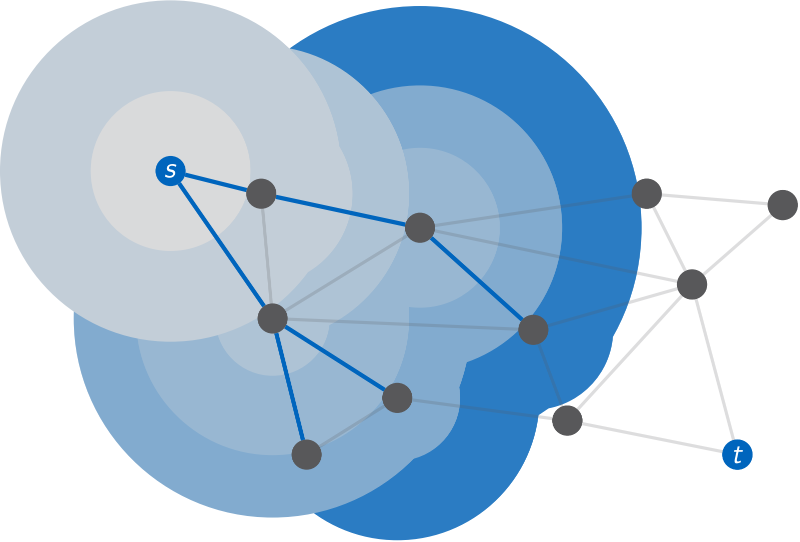
s
t

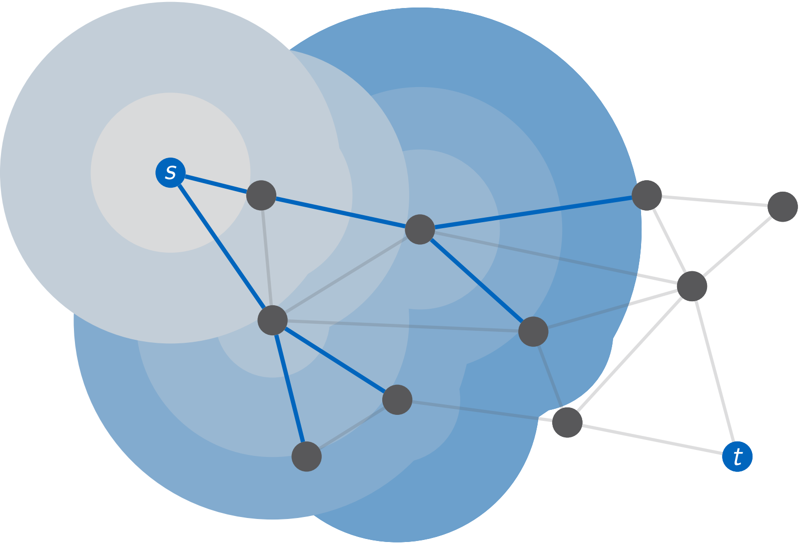

s
t

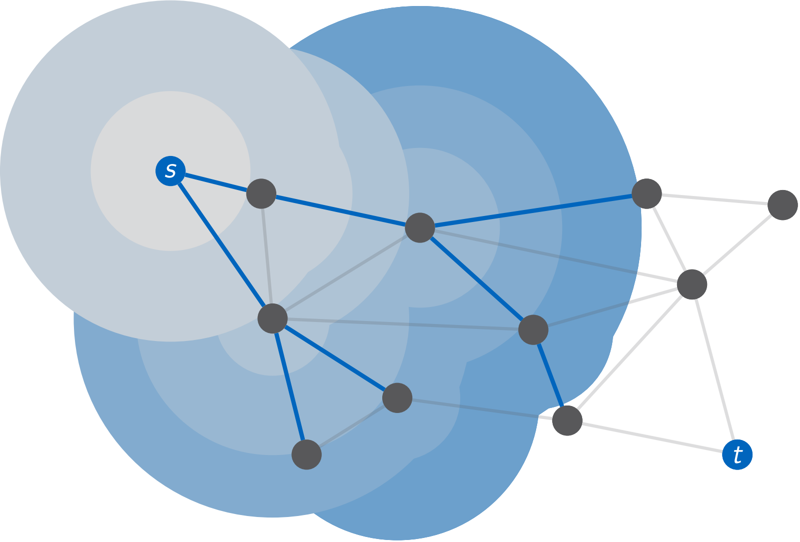

s
t

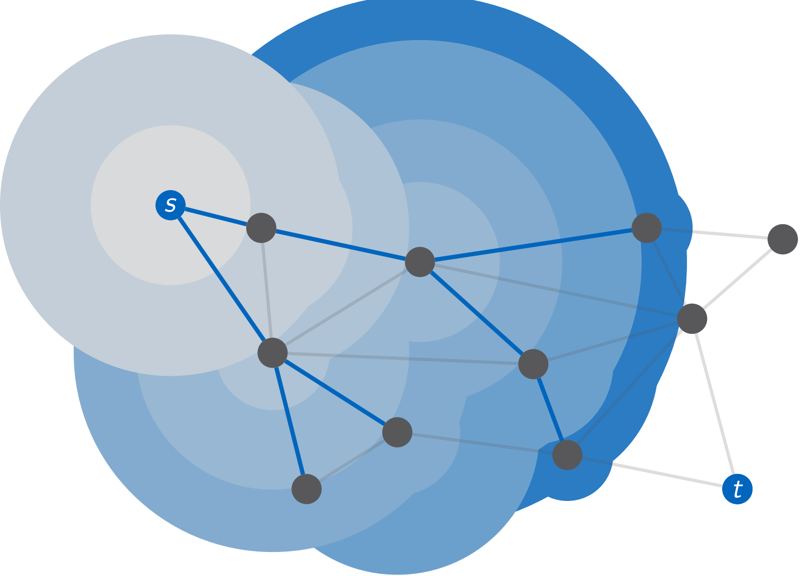


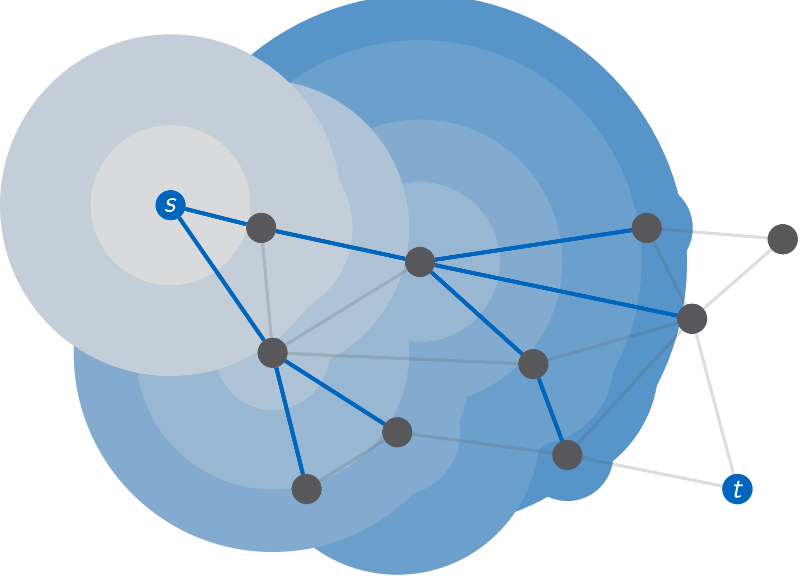
s
t

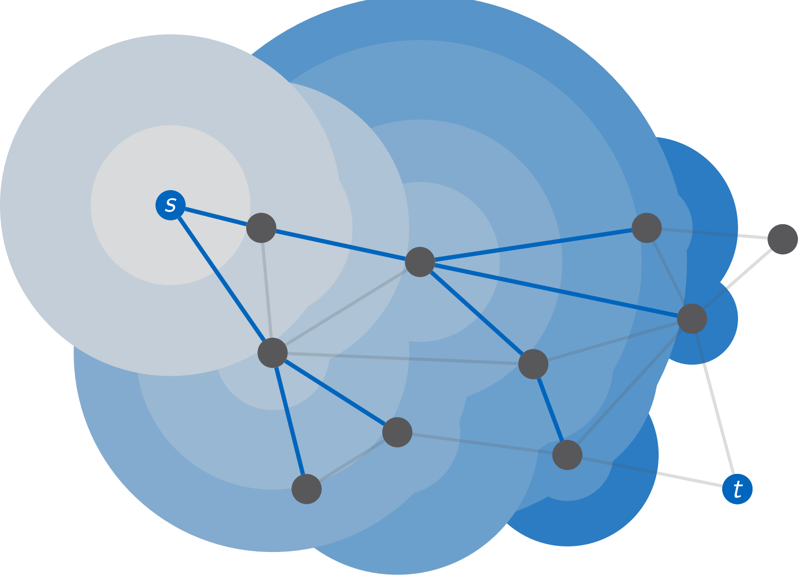

s
t

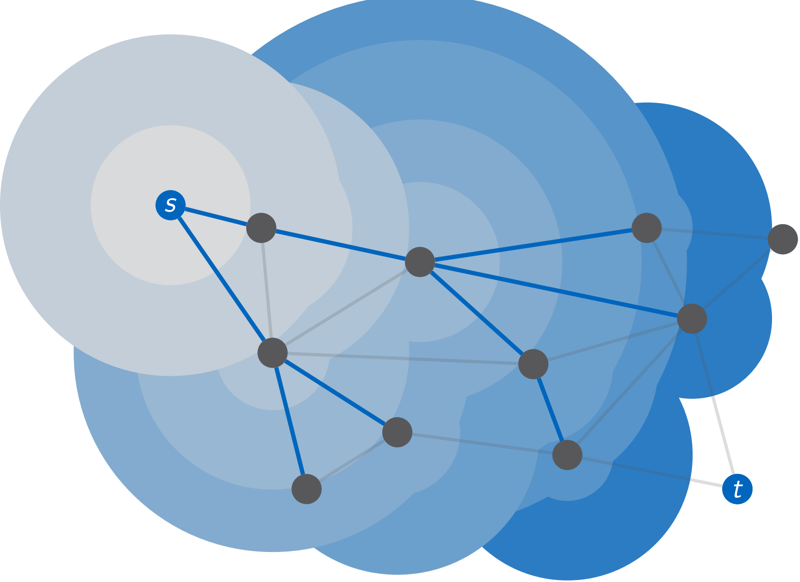

s
t

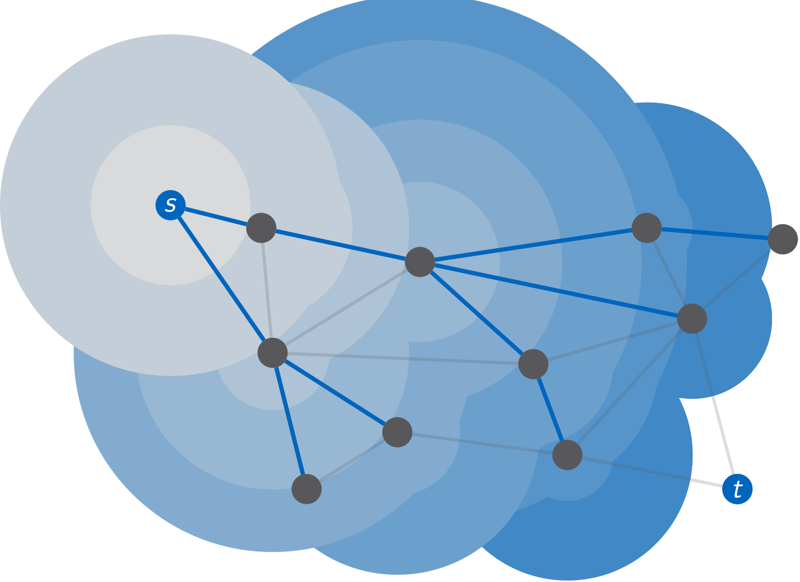

s
t

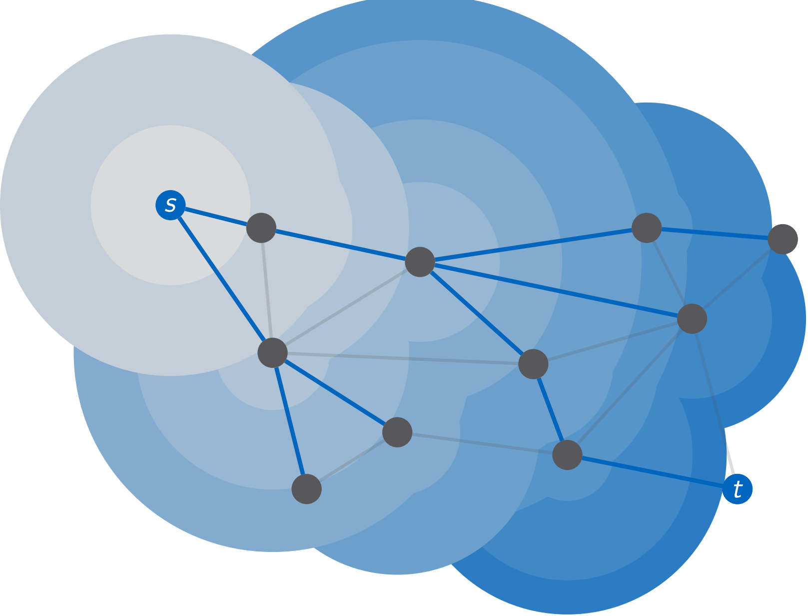

s
t

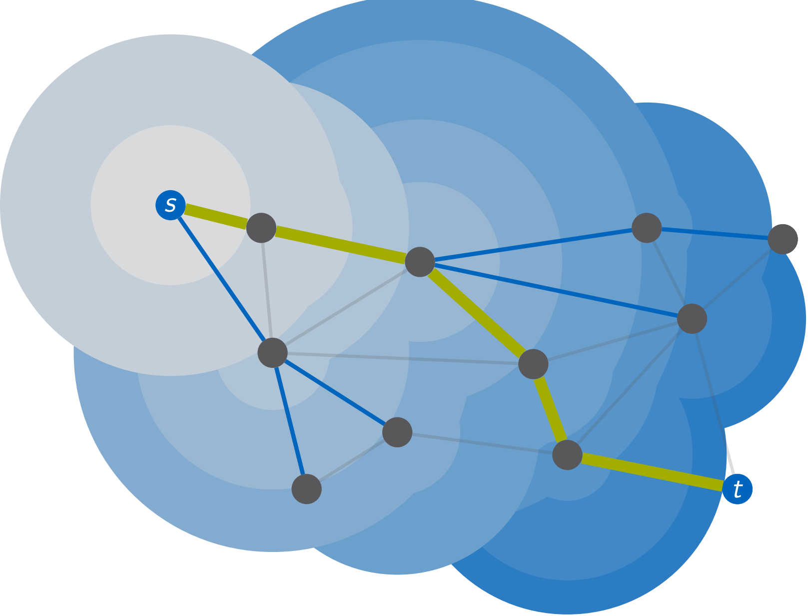

s
t


s
t


s
t


s
t


s
t


s
t

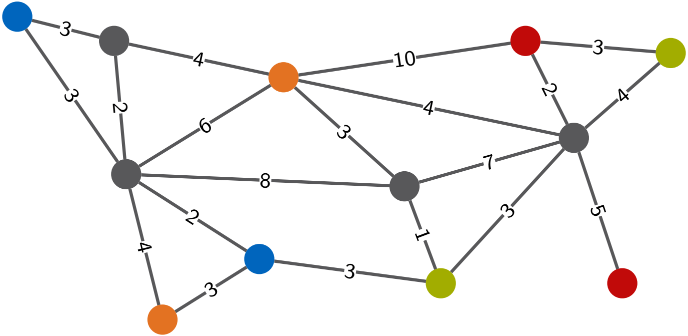
The primal-dual method Example II: The generalized Steiner tree problem

(aka Steiner Forest)

Steiner Forest

Input: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_+$,
terminal pairs $\{s_i, t_i\}$ for $i \in [n]$

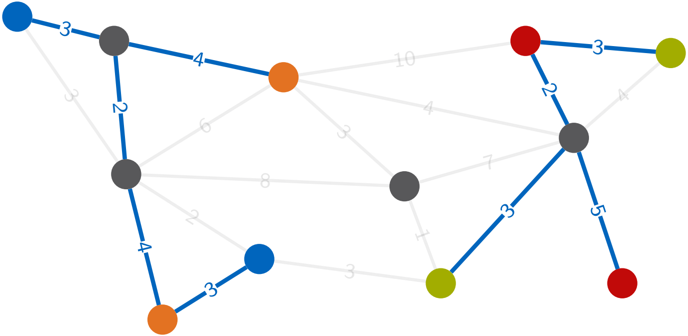
Task: find $F \subseteq E$ containing an s_i-t_i-path for every $i \in [n]$
minimizing $\sum_{e \in F} w_e$



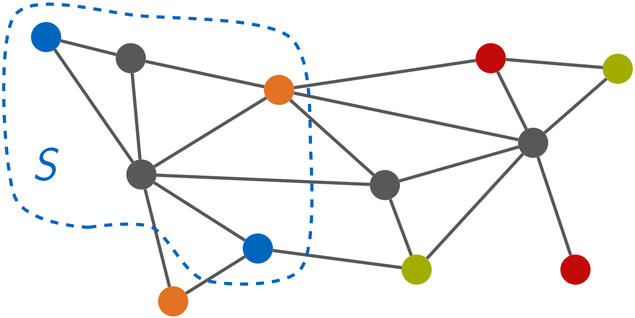
Steiner Forest

Input: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_+$,
terminal pairs $\{s_i, t_i\}$ for $i \in [n]$

Task: find $F \subseteq E$ containing an s_i-t_i-path for every $i \in [n]$
minimizing $\sum_{e \in F} w_e$



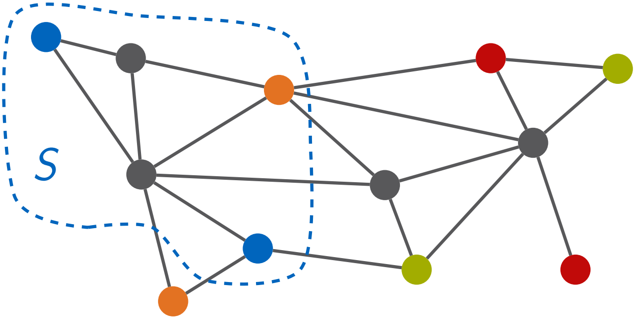
LP relaxation



$$\mathcal{F} := \{S \subseteq V \;:\; |S \cap \{s_i, t_i\}| = 1 \text{ for some } i \in [n]\}$$

$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e \;\geq\; 1 \qquad \forall\, S \in \mathcal{F} \\ & x_e \;\in \{0,1\} \qquad \forall\, e \in E \end{aligned}$$

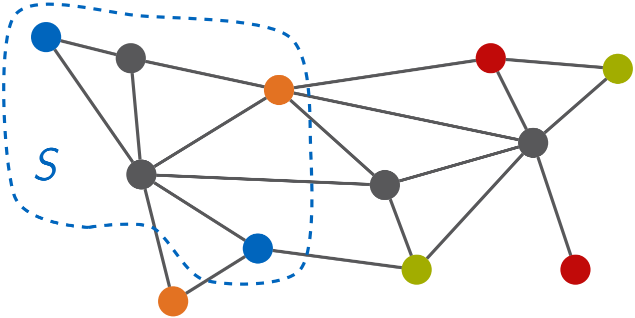
LP relaxation



$$\mathcal{F} := \{S \subseteq V \,:\, |S \cap \{s_i, t_i\}| = 1 \text{ for some } i \in [n]\}$$

$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e \geq 1 \qquad \forall\, S \in \mathcal{F} \\ & x_e \in \{0,1\} \qquad \forall\, e \in E \end{aligned}$$

LP relaxation



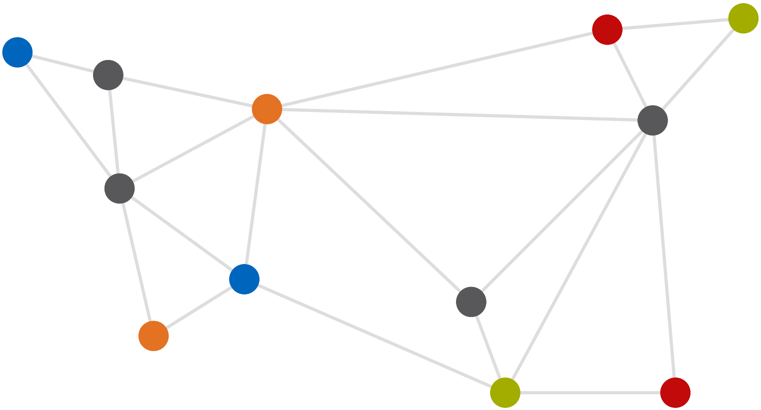
$$\mathcal{F} := \{S \subseteq V \ : \ |S \cap \{s_i, t_i\}| = 1 \text{ for some } i \in [n]\}$$

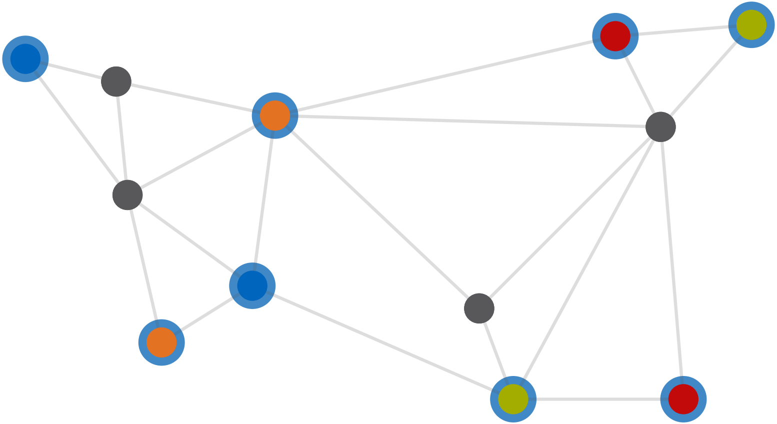
$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e \geq 1 \quad \forall\, S \in \mathcal{F} \\ & x_e \geq 0 \quad \forall\, e \in E \end{aligned}$$

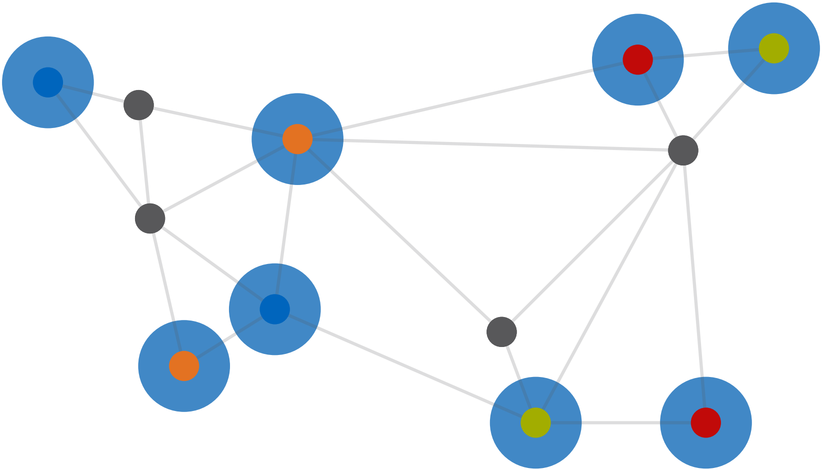
Primal-dual algorithm

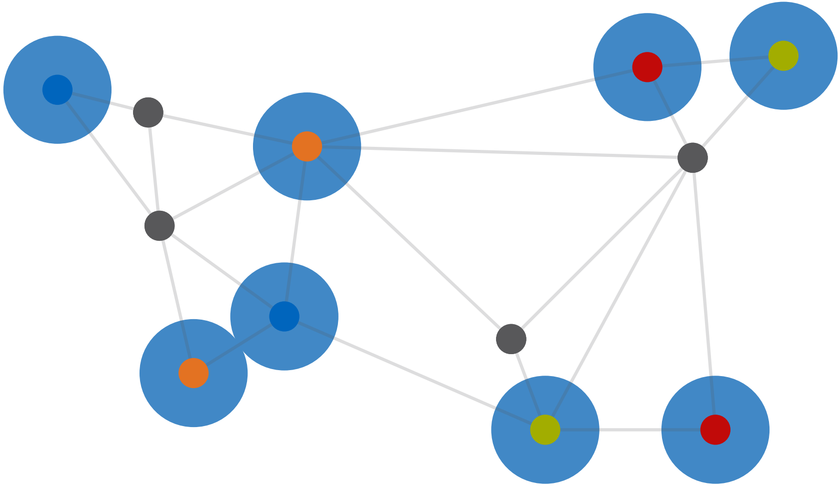
Algorithm

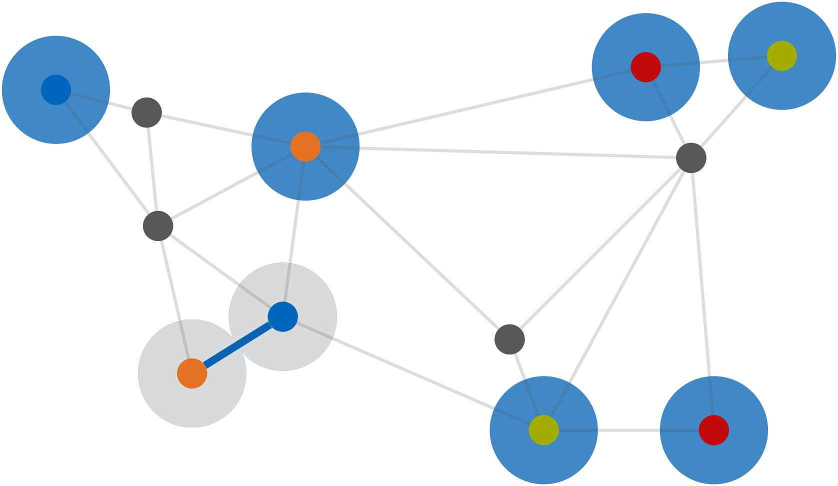
1. $F := \emptyset$, $y := 0$
2. while $\exists\, i : F$ does not contain s_i-t_i-path
   - $\mathcal{C} := \{C \text{ conn. comp. of } (V, F) \,:\, |C \cap \{s_i, t_i\}| = 1 \text{ for some } i\}$
   - Uniformly increase y_C for all $C \in \mathcal{C}$ until $\sum_{S \in \mathcal{F}_e} y_S = w_e$ for some $e \in \bigcup_{C \in \mathcal{C}} \delta(C)$. (increase can be 0)
   - $F := F \cup \{e\}$
3. $F := F \setminus \{e \in F \,:\, e \text{ is on no } s_i\text{-}t_i\text{-path in } F\}$
4. return F

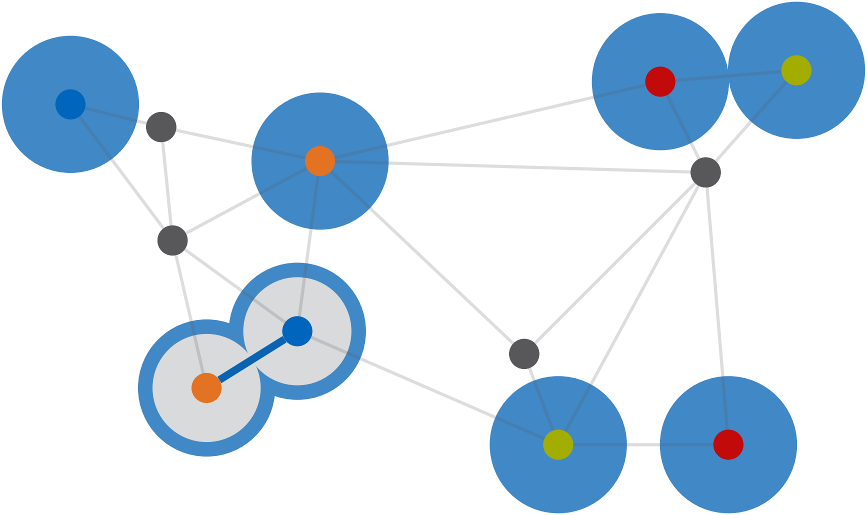


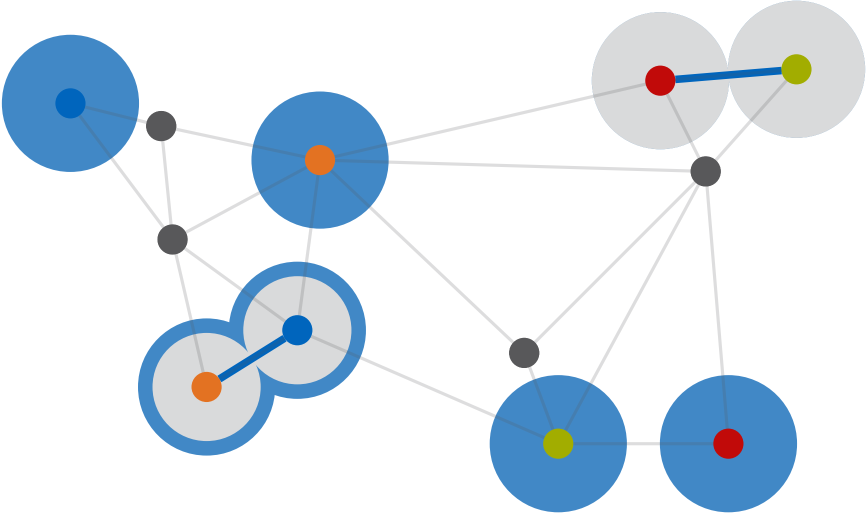


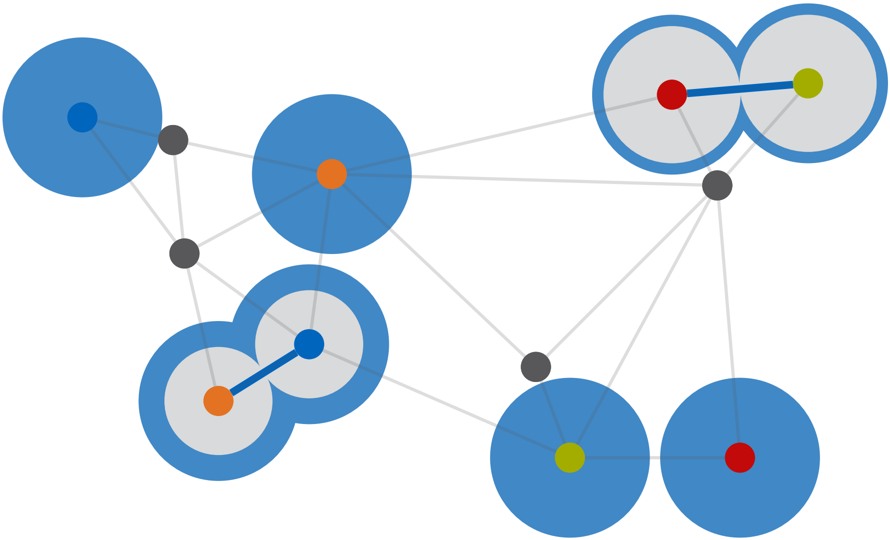


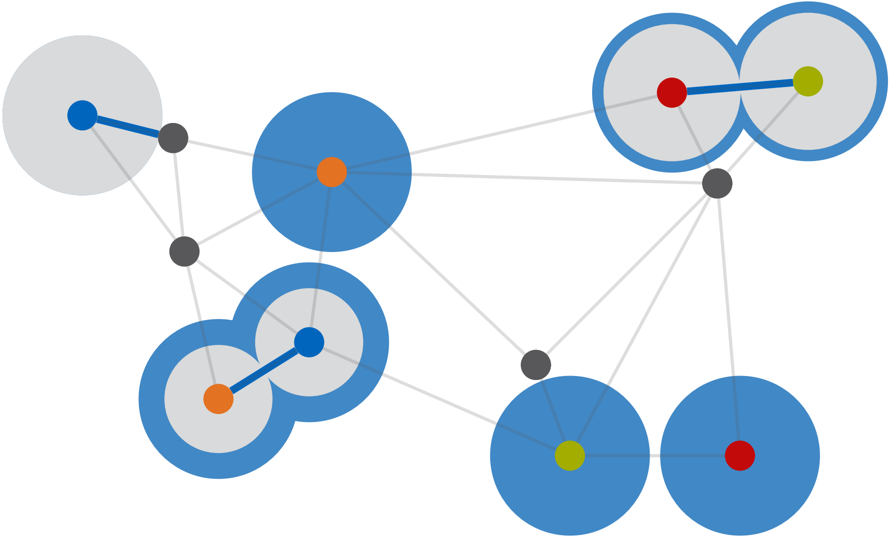


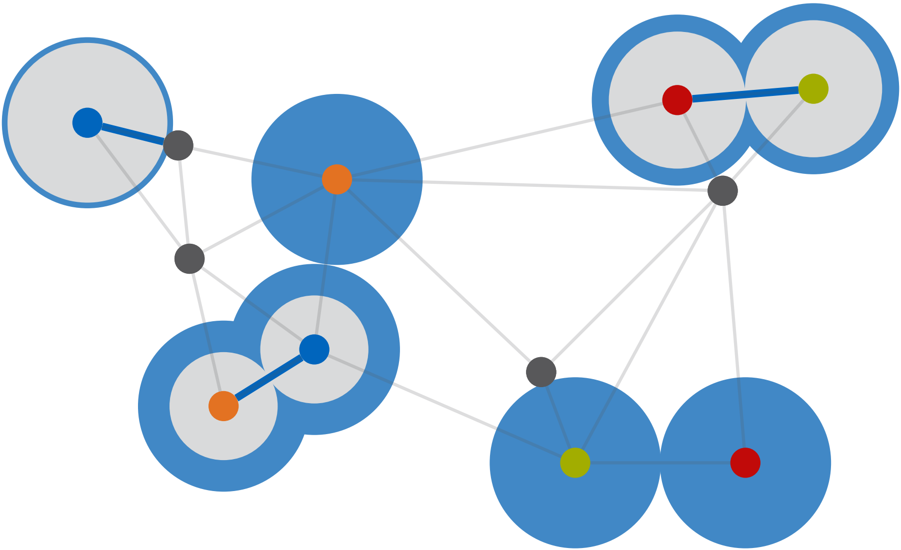


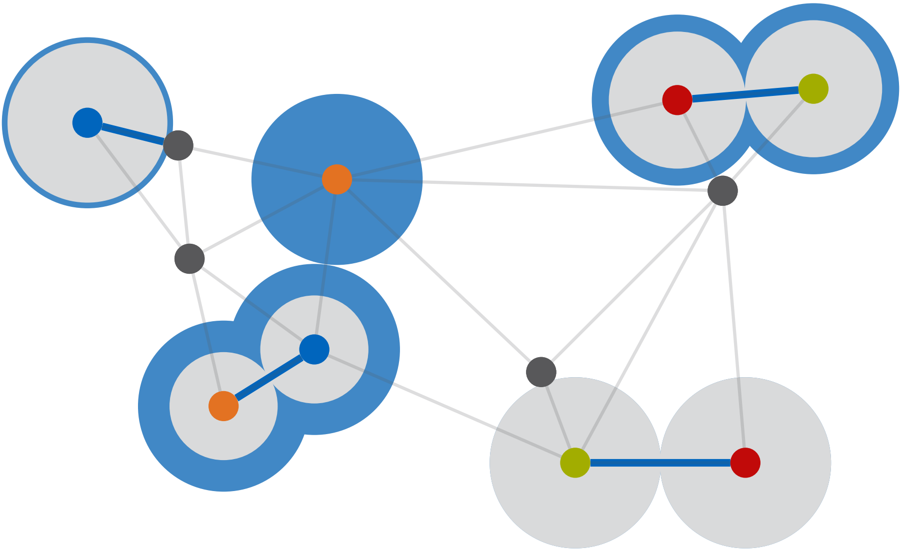


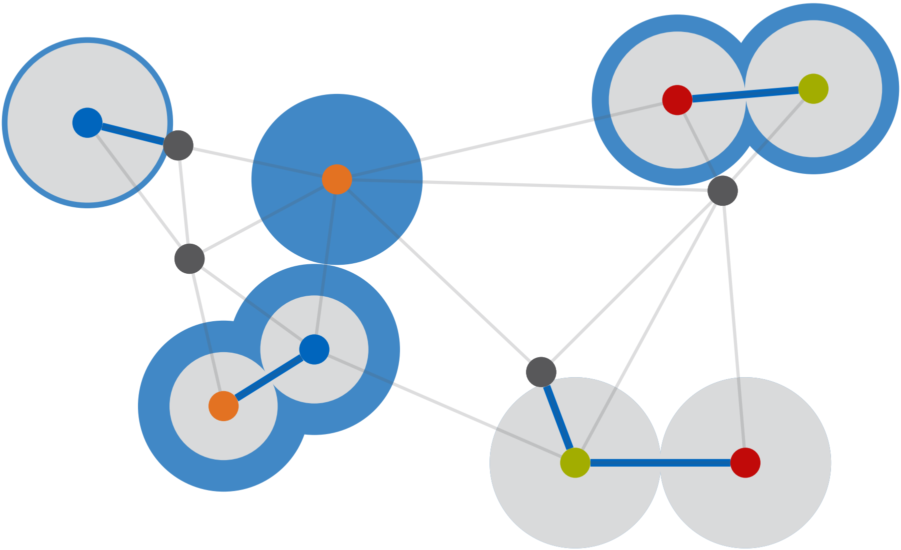


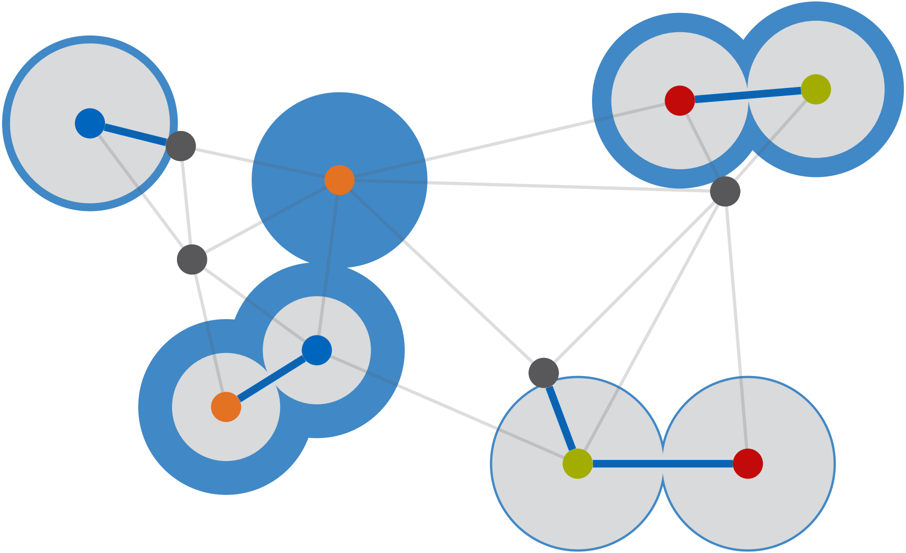


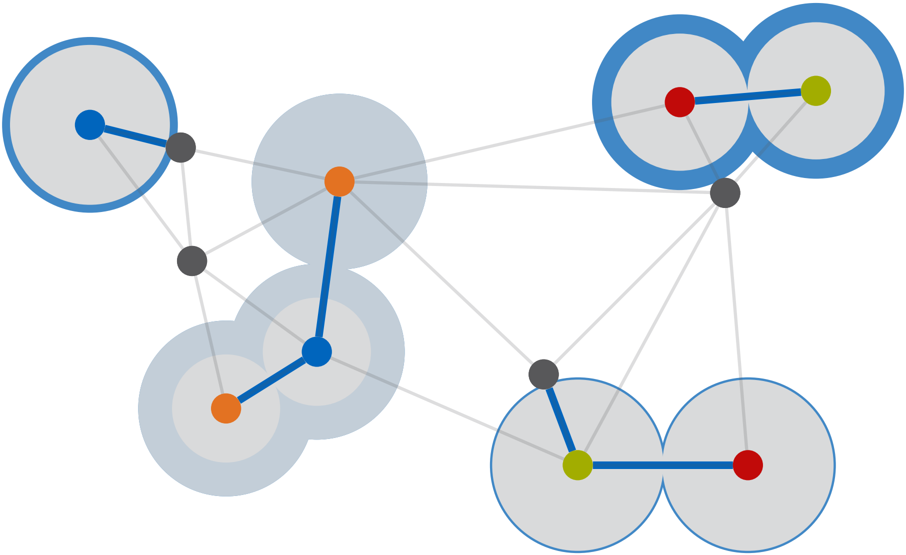


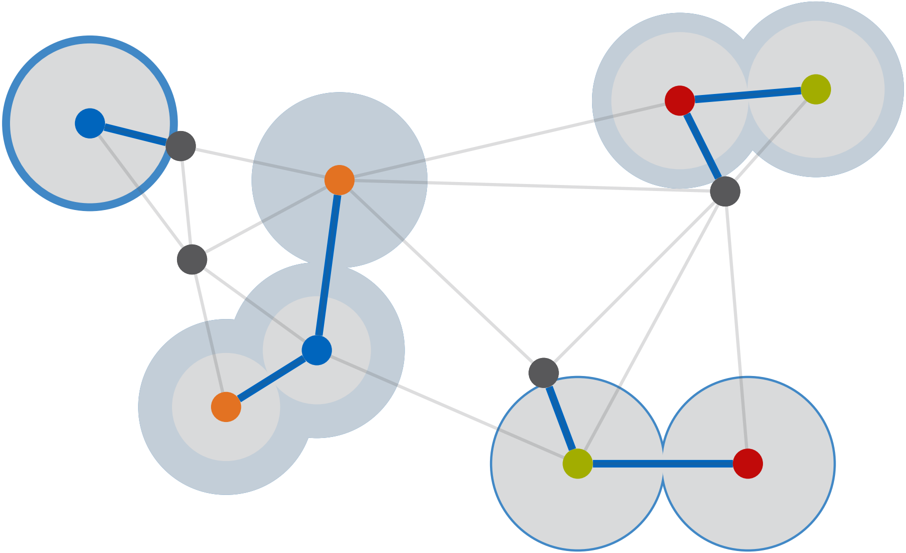


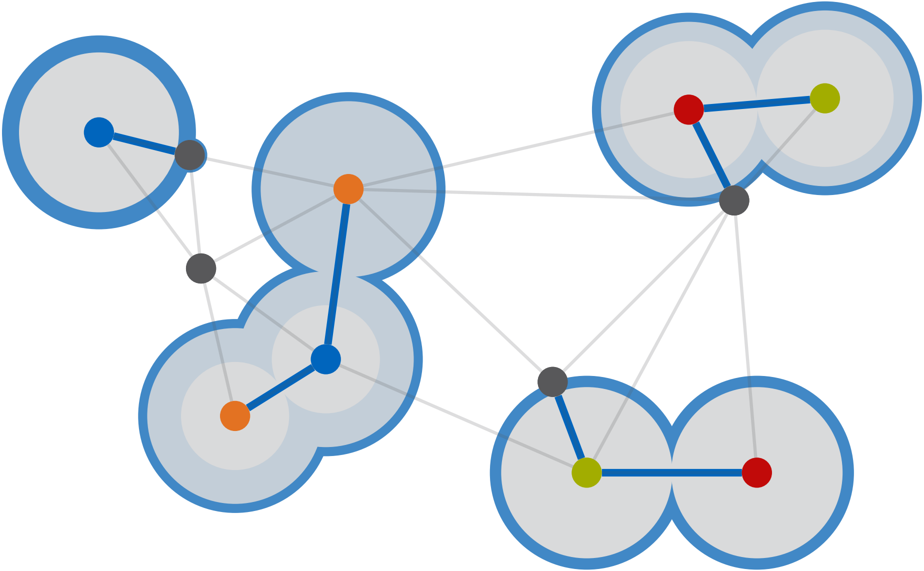


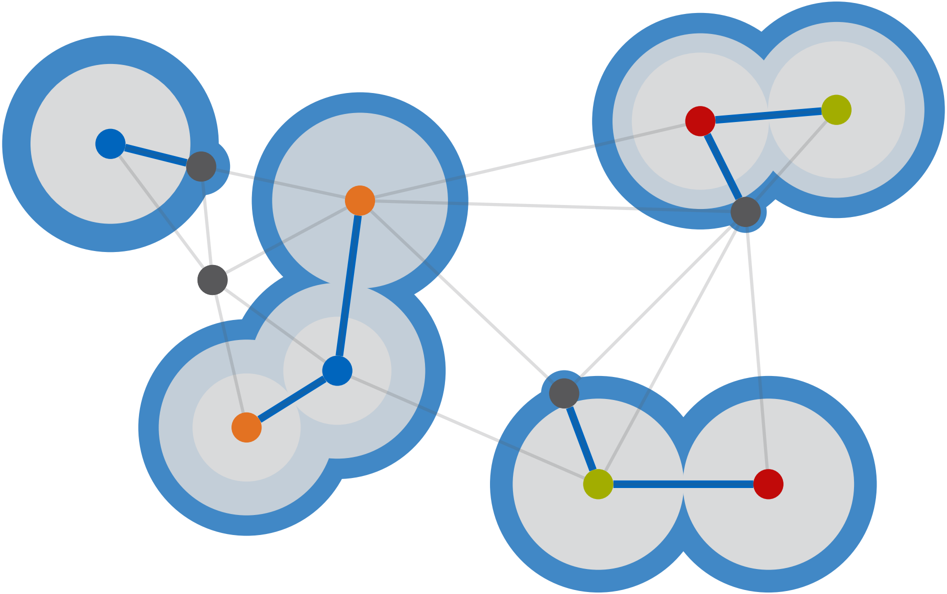


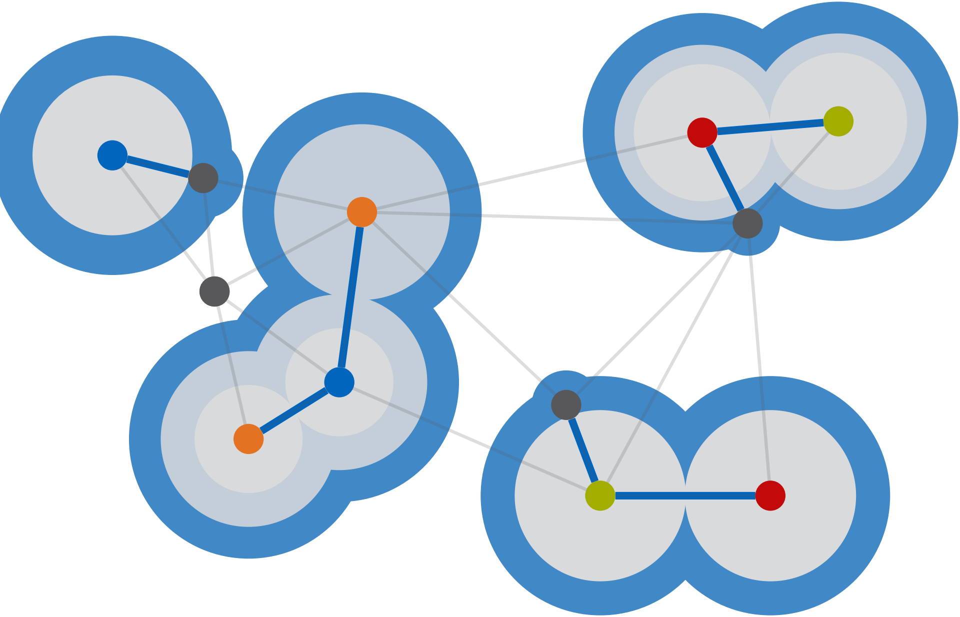


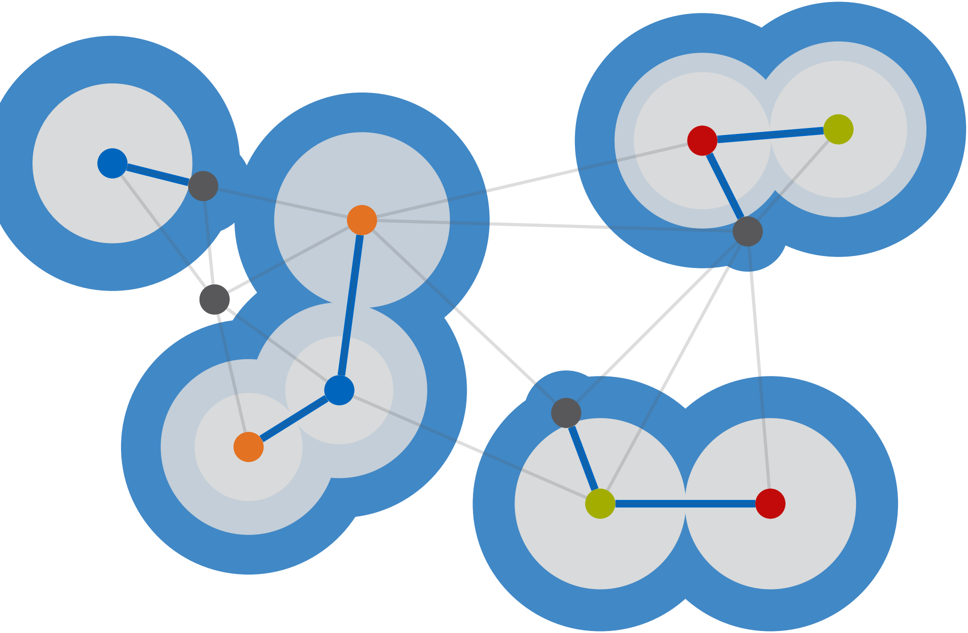


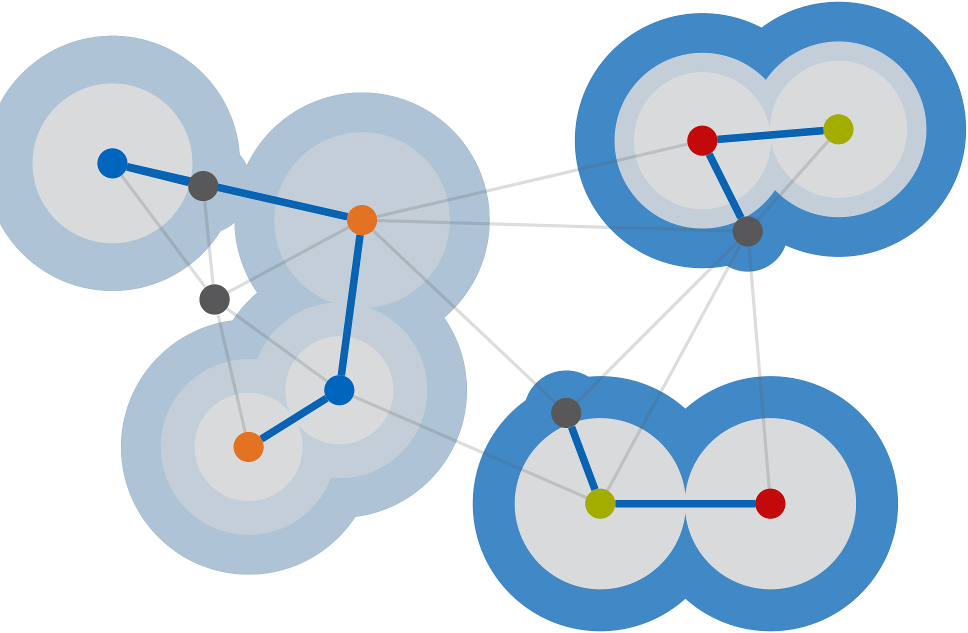


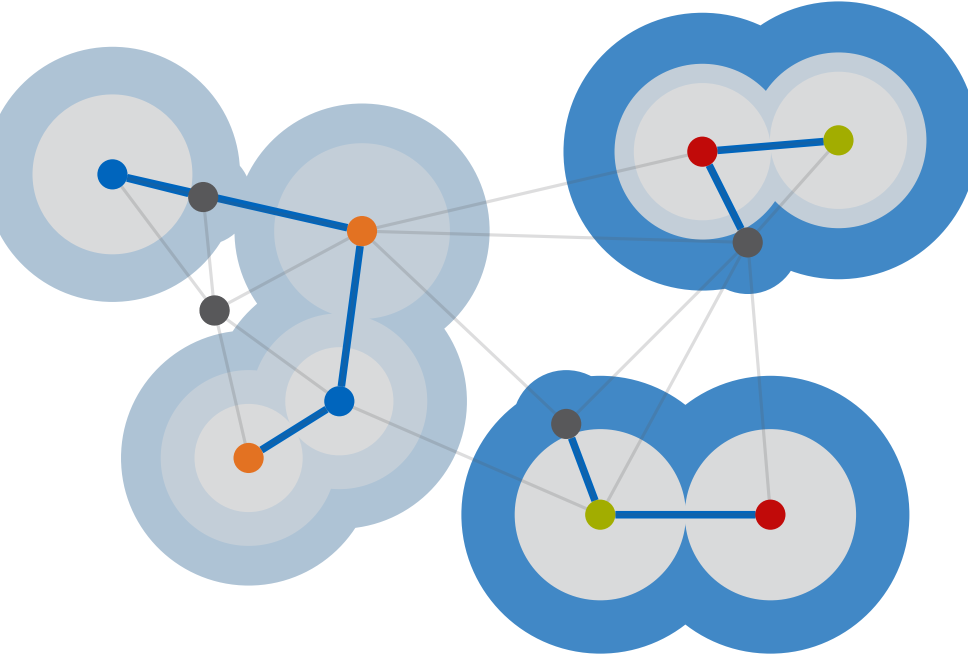


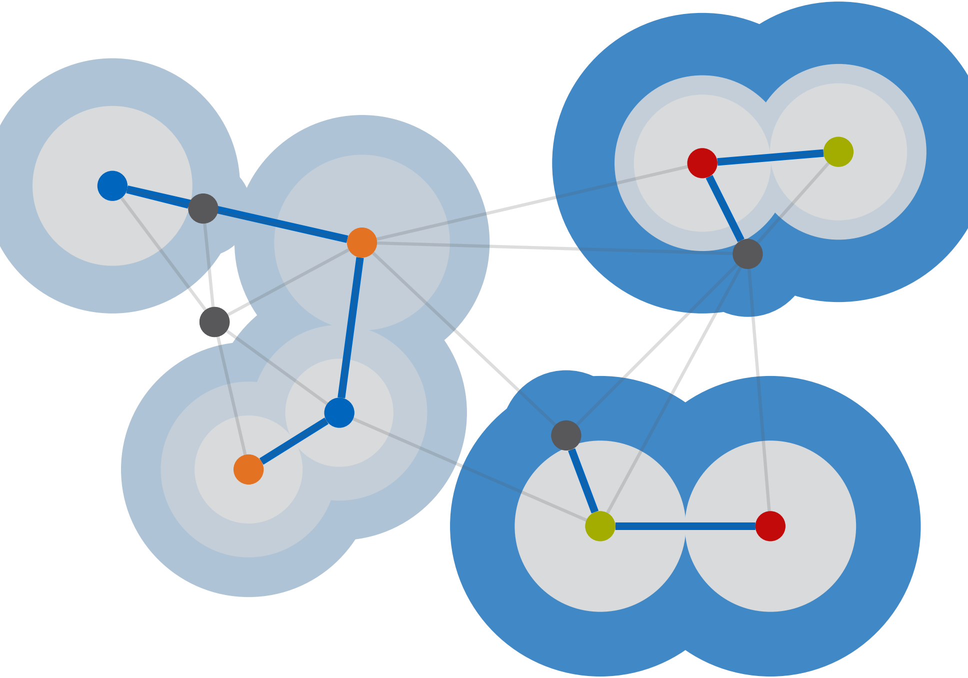


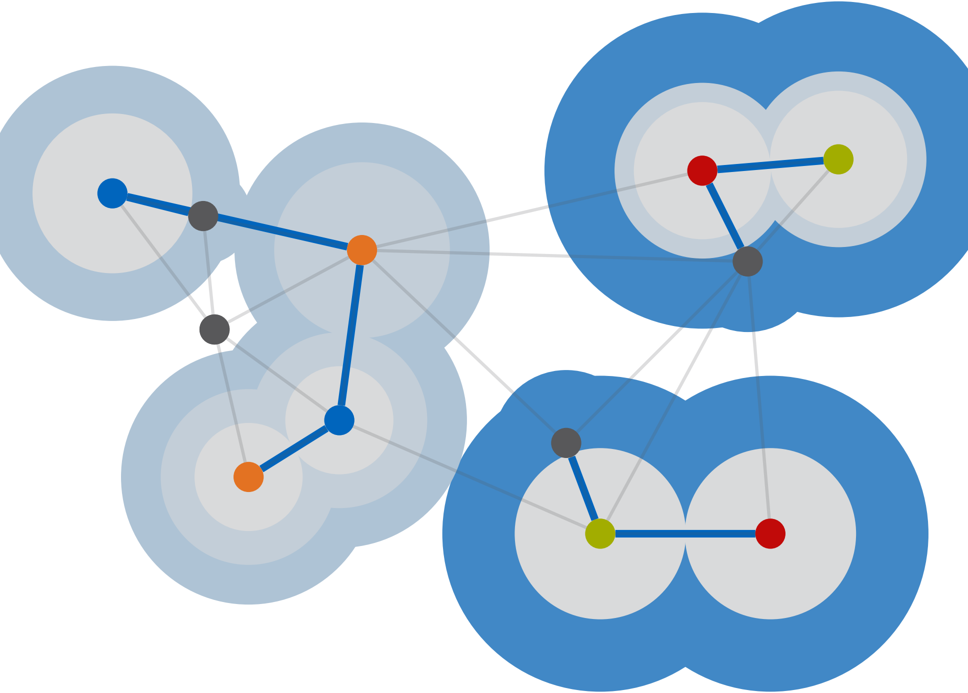


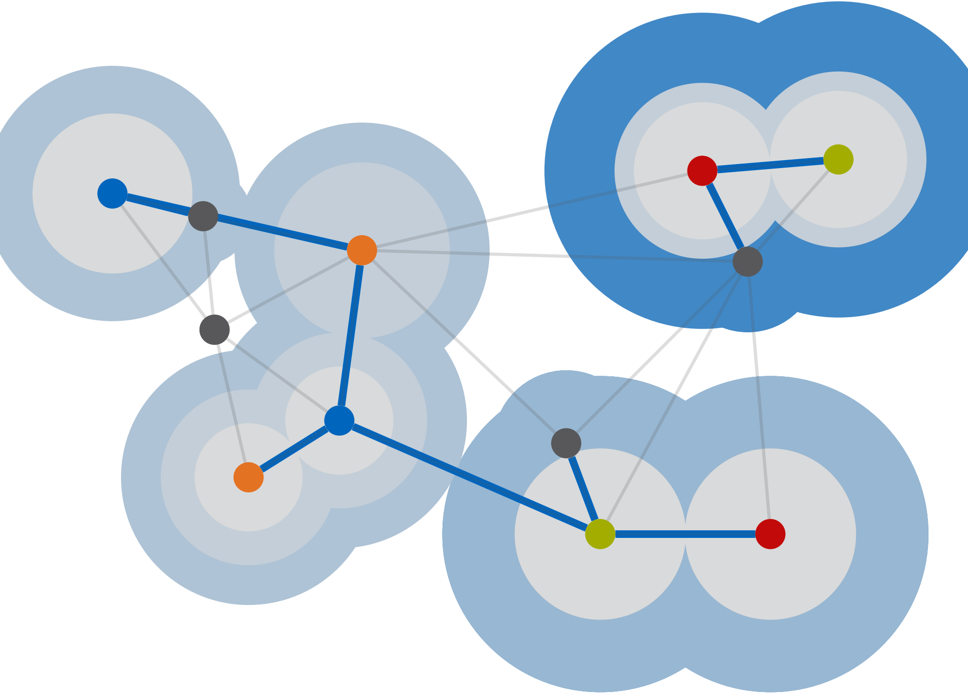


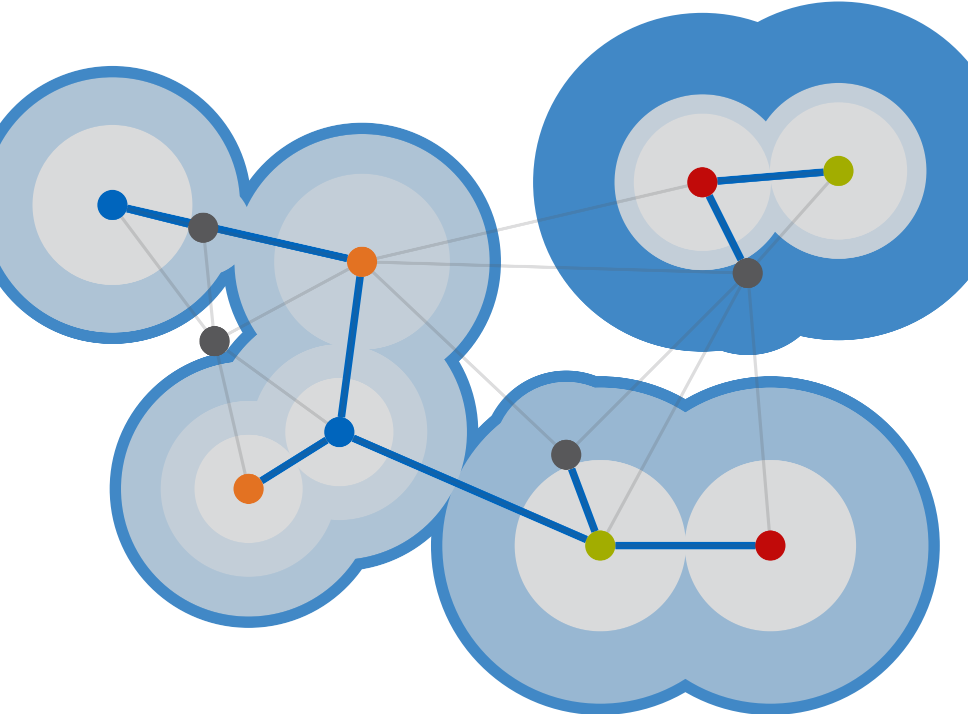


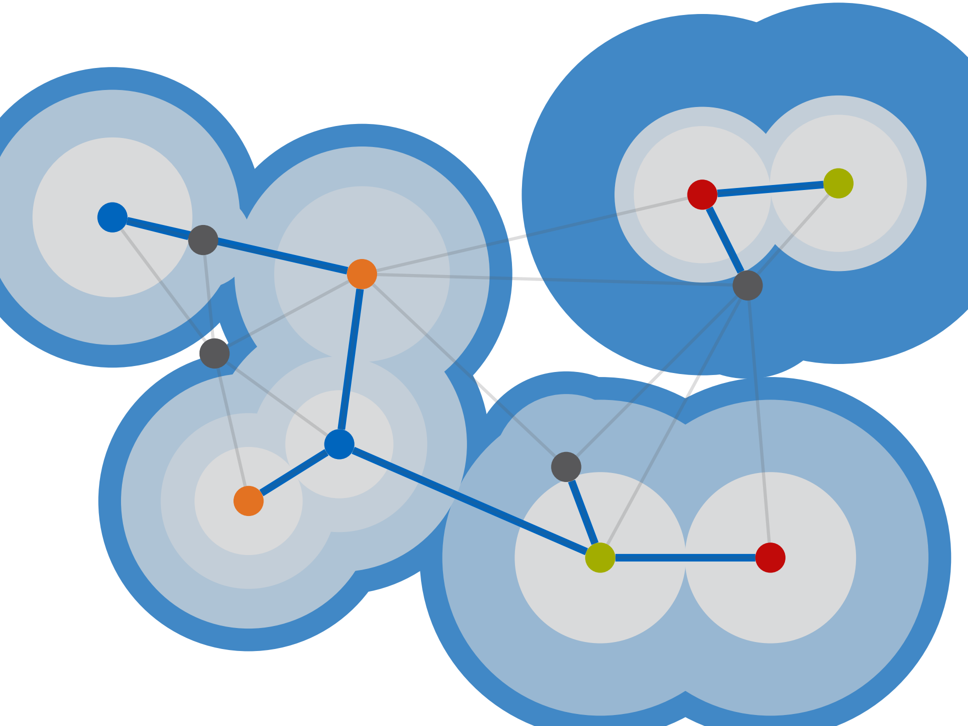


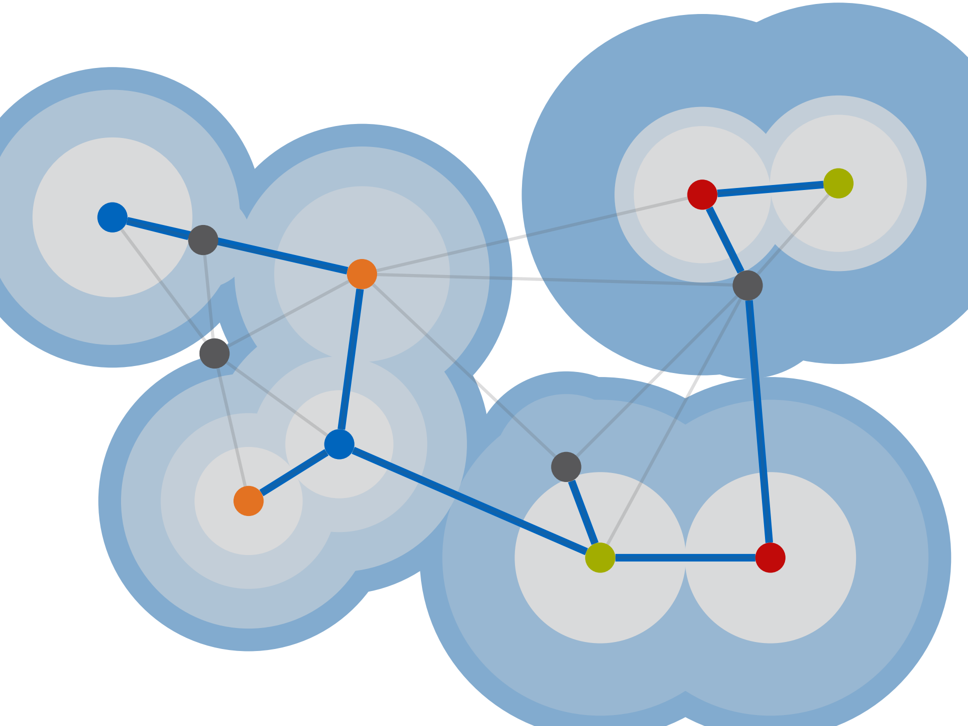


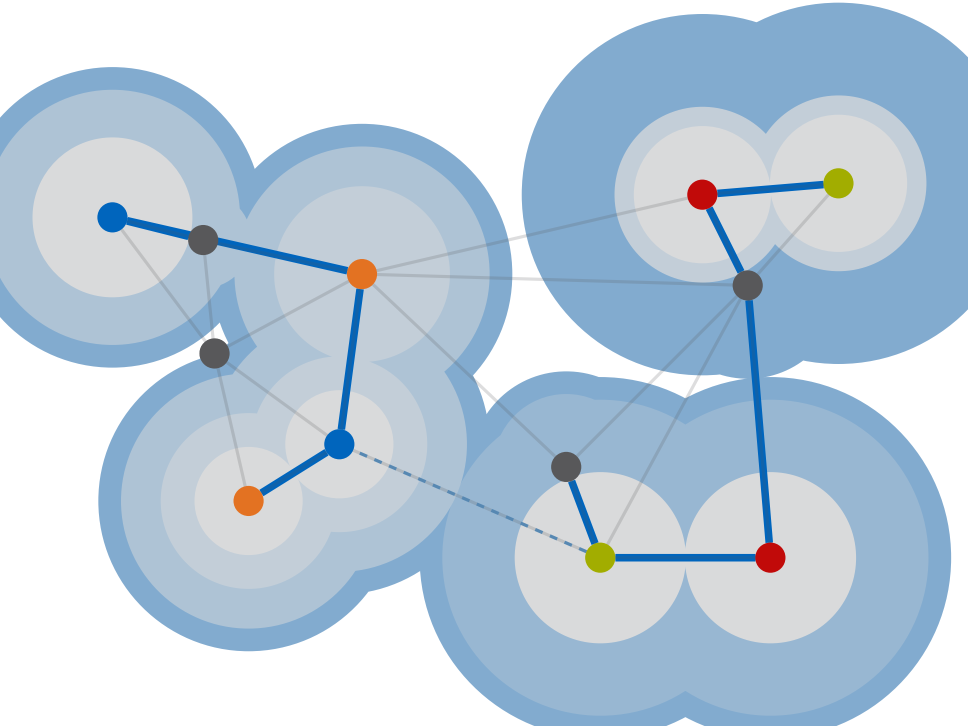


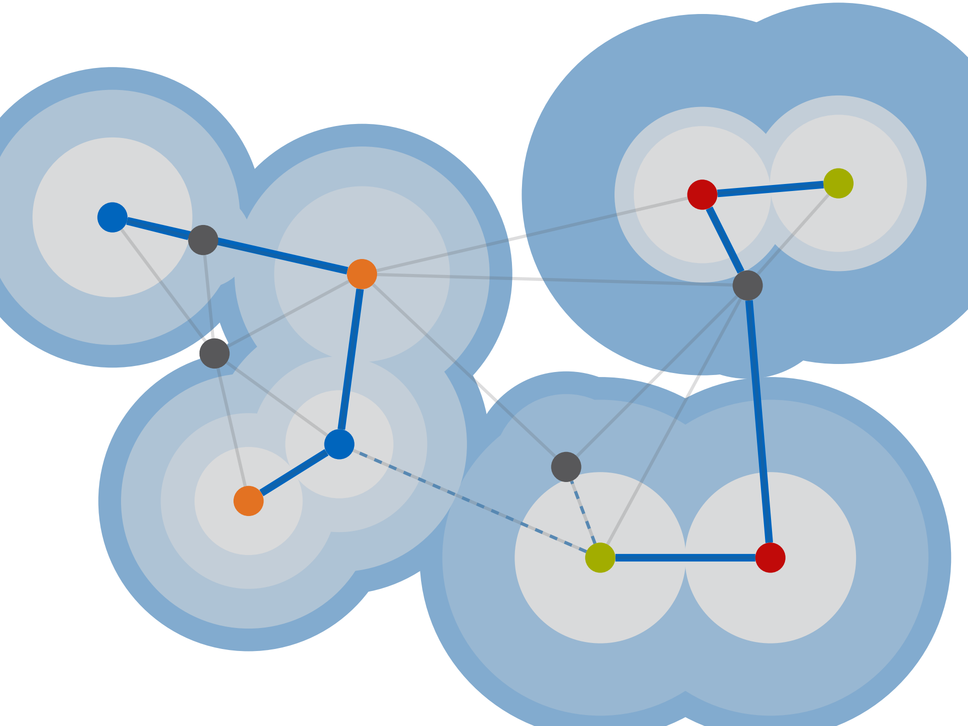












Primal-dual algorithm

Algorithm

1. while $\exists\, i : F$ does not contain s_i-t_i-path
   - $\mathcal{C} := \{C \text{ conn. comp. of } (V, F) : |C \cap \{s_i, t_i\}| = 1 \text{ for some } i\}$
   - Uniformly increase y_C for all $C \in \mathcal{C}$ until $\sum_{S \in \mathcal{F}_e} y_S = w_e$ for some $e \in \bigcup_{C \in \mathcal{C}} \delta(C)$. (increase can be 0)
   - $F := F \cup \{e\}$
2. $F := F \setminus \{e \in F : e \text{ is on no } s_i\text{-}t_i\text{-path in } F\}$
3. return F

Theorem 9.2

Primal-dual is a 2-approximation algorithm for Steiner Forest.

Lemma 9.3

Let $\bar{F}$ be the tree returned by the alrogithm. At the beginning of every iteration: $\sum_{C \in \mathcal{C}} |\bar{F} \cap \delta(C)| \leq 2|\mathcal{C}|$