22.01.23

Lemma: We may assume $(s,t) \notin E$.

~~Definition:~~ We now define the merge update flow network which, given an update flow network $G=(V,E,\mathcal{P},s,t,c)$ and two flow pairs $P,P' \in \mathcal{P}$, tries to merge P and P' into a single flow pair $\tilde{P}$ by ~~appending~~ concatenating and P' to P. Intuitively, we want to set $\tilde{P} := P \cup P'$ and add edge (t,s) to both $\tilde{P}^o$ and $\tilde{P}^u$. As this would not only add an outgoing edge to t and an incoming edge to s but also introduce a cycle (from s to t to s) into $\tilde{P}$, the construction of the merge update flow network is a bit more involved.

Either we F, F' instead of P, P' or we $\bigcup_{F\in\mathcal{P}}$ below.

I probably don't want to have a definition for this after all and instead put everything into the proof of the lemma.

Definition: Given ~~a flow~~ an update flow network $G=(V,E,\mathcal{P},s,t,c)$ and two flow pairs $P,P'\in\mathcal{P}$, ~~a~~ the merge update flow network $\tilde{G}=(\tilde{V},\tilde{E},\tilde{\mathcal{P}},s,t,\tilde{c})$ of G w.r.t. P and P' is defined as follows.

~~Let u_{po}, u_{pu}~~ As an intermediate step, we define the update flow network $\hat{G}=(\hat{V},\hat{E},\hat{\mathcal{P}},s,t,\hat{c})$, which replaces the last edge on P^o, P^u ~~an~~ and the first edge on P'^o, P'^u with two edges respectively, as follows. Let $u_{po}, u_{pu}, v_{p'o}, v_{p'u}$ be such that $(u_{po},t)\in E(P^o)$, $(u_{pu},t)\in E(P^u)$,
($u_{po}=u_{pu}$, $v_{p'o}=v_{p'u}$ may hold)

More assumption? $(s,v_{p'o})\in E(P'^o)$, $(s,v_{p'u})\in E(P'^u)$. By lemma ~, we may assume $u_{po}, u_{pu}, v_{p'o}, v_{p'u} \notin \{s,t\}$.*

We introduce ~~two~~ four new vertices $\tilde{u}_{po}, \tilde{u}_{pu}, \tilde{v}_{p'o}, \tilde{v}_{p'u}$.

We replace edges $(u_{po},t), (u_{pu},t), (s,v_{p'o}), (s,v_{p'u})$ with the following edges:

- $(\tilde{u}_{po},t), (\tilde{u}_{pu},t), (s,\tilde{v}_{p'o}), (s,\tilde{v}_{p'u})$ with capacity $\sum_{P\in\mathcal{P}} d_P$
- $(u_{po},\tilde{u}_{po})$ with capacity $c(u_{po},t)$
- $(u_{pu},\tilde{u}_{pu})$ $c(u_{pu},t)$
- $(\tilde{v}_{p'o},v_{p'o})$ $c(s,v_{p'o})$
- $(\tilde{v}_{p'u},v_{p'u})$ $c(s,v_{p'u})$

We update all flow pairs to use the new edges instead of the old edges, that is, for every flow pair $P\in\mathcal{P}$, we define $\hat{P}$ as follows:

$\hat{E}(\hat{P}^o) = E(P^o) - \{(u_{po},t),(u_{pu},t),(s,v_{p'o}),(s,v_{p'u})\} \cup$

$\{(u_{po},\tilde{u}_{po}),(\tilde{u}_{po},t) : (u_{po},t)\in E(P^o)\} \cup$

$\{(u_{pu},\tilde{u}_{pu}),(\tilde{u}_{pu},t) :$

$\{\cancel{(v_{p'o},v_{p'u})}\ (s,\tilde{v}_{p'o}),(\tilde{v}_{p'o},v_{p'o}) :$

$\{(s,\tilde{v}_{p'u}),(\tilde{v}_{p'u},v_{p'u}) :$

$\hat{E}(\hat{P}^u)$ is defined analogously.

BRUNNEN

We are now ready to specify intermediate update flow network $\hat{G}$. ~~$\hat{c}$ is defined as above for the new edges~~ $\hat{c}(e)$ is defined as above if e is a new edge and equal to $c(e)$ otw. ~~$\hat{V}(\hat{P}) = V(\hat{E}(\hat{P}))$ for all~~

$\hat{\mathcal{P}} := \{\hat{P} : P \in \mathcal{P}\}$, where $\hat{V}(\hat{P}) = \hat{V}(\hat{E}(\hat{P}))$. $d_{\hat{P}} = d_P$ $\hat{V} := \bigcup_{\hat{P} \in \hat{\mathcal{P}}} \hat{V}(\hat{P})$. $\hat{E} := \bigcup_{\hat{P} \in \hat{\mathcal{P}}} \hat{E}(\hat{P})$.

We now concatenate $\hat{P}$ and $\hat{P}'$ by connecting $\tilde{u}_{p^o}$ to $\tilde{v}_{p'^o}$ and $\tilde{u}_{p^u}$ to $\tilde{v}_{p'^u}$.

* We want to connect u_{p^o} to $v_{p'^o}$ and u_{p^u} to $v_{p'^u}$. Note, however, that we may not in general connect them via a single edge, respectively, as this may not only change congestion constraints on ← Elaborate edges $(u_{p^o}, t), (u_{p^u}, t), (s, v_{p'^o}), (s, v_{p'^u})$ but ~~may~~ also union blocks $b(u_{p^o}, P), b(v_{p'^o}, P')$ or $b(u_{p^u}, P), b(v_{p'^u}, P')$. Elaborate

As ~~noted~~ indicated above, we may not in general connect them via a single edge, respectively. Therefore, we introduce another vertex w and edges $(\tilde{u}_{p^o}, w), (\tilde{u}_{p^u}, w), (w, \tilde{v}_{p'^o}), (w, \tilde{v}_{p'^u})$ with capacity $\sum_{\hat{P} \in \hat{\mathcal{P}}} \hat{d}_{\hat{P}}$.

Moreover, we replace flow pair $\hat{P}, \hat{P}'$ with a new flow pair $\tilde{P}$ and set

$\tilde{E}(\tilde{P}^o) := \hat{E}(\hat{P}^o) - \{(\tilde{u}_{p^o}, t)\} \cup \{(\tilde{u}_{p^o}, w)\} \cup \hat{E}(\hat{P}'^o) - \{(s, \tilde{v}_{p'^o})\} \cup \{(w, \tilde{v}_{p'^o})\}$. $\tilde{E}(\tilde{P}^u)$ is defined analogously.

We are finally ready to specify merge update flow network $\tilde{G}$.

$\tilde{c}(e)$ is defined as above if e is a new edge and equal to $\hat{c}(e)$ otw. $\tilde{\mathcal{P}} := \hat{\mathcal{P}} - \{\hat{P}, \hat{P}'\} \cup \{\tilde{P}\}$, where

$\tilde{V}(\tilde{P}) := \tilde{V}(\tilde{E}(\tilde{P}))$ and $d_{\tilde{P}} = d_{\hat{P}} = d_{\hat{P}'}$. $\tilde{V} := \bigcup_{\tilde{P} \in \tilde{\mathcal{P}}} \tilde{V}(\tilde{P})$. $\tilde{E} := \bigcup_{\tilde{P} \in \tilde{\mathcal{P}}} E(\tilde{P})$. ← $\max\{d_{\hat{P}}, d_{\hat{P}'}\}$

<u>Lemma:</u> Let G be an update flow network and $F, F' \in \mathcal{P}$ be two flow pairs such that F and F' have no common vertices other than s, t and $d_P = d_{P'}$. Then there is a feasible block sequence $B = (b_1, \ldots, b_l)$ for G iff there is a feasible block sequence $\tilde{B} = (\tilde{b}_1, \ldots, \tilde{b}_l)$ for ~~over~~ the merge update flow network $\tilde{G}$ of G w.r.t. F and F'.

<u>Proof.</u>

<u>"⇒"</u>

Let $B = (b_1, \ldots, b_l)$ be a feasible block sequence for G. We construct a feasible block sequence $\tilde{B} = (\tilde{b}_1, \ldots, \tilde{b}_l)$ for $\tilde{G}$ as follows. The idea is to map every block $\tilde{b} \in \tilde{B}$ to a block $f(\tilde{b}) \in B$ such that $\tilde{b} \in \tilde{b}_i$ iff $f(\tilde{b}) \in b_i$. We define mappings $f_V : \tilde{V} \times \tilde{\mathcal{P}} \to V$, $f_{\mathcal{P}} : \tilde{V} \times \tilde{\mathcal{P}} \to \mathcal{P}$, $f : \tilde{V} \times \tilde{\mathcal{P}} \to V \times \mathcal{P}$, $f_B : \tilde{B} \to B$ as follows:

$$f_V(\tilde{v}, \tilde{P}) = \begin{cases} u_{p^o} & \text{if } \tilde{v} = \tilde{u}_{p^o} \\ u_{p^u} & \text{if } \tilde{v} = \tilde{u}_{p^u} \\ s & \text{if } \tilde{v} = \tilde{v}_{p'^o} \text{ or } \tilde{v} = \tilde{v}_{p'^u} \text{ or } \tilde{v} = w \\ v & \text{otw. } [\tilde{v} = v] \end{cases}$$

Remove argument $\tilde{P}$.

22.01.23

Proof (ctd.)

Is this even well-defined? What if $f_v(\tilde{v}, \tilde{P}) = s$? given we need to $\tilde{x}$ remove argument. Remove?

$$f_p(\tilde{v}, \tilde{P}) = \begin{cases} F & \text{if } \tilde{P} = \tilde{F} \text{ and } f_v(\tilde{v}, \tilde{P}) \in V(F) \\ F' & \text{if } \tilde{P} = \tilde{F} \text{ and } f_v(\tilde{v}, \tilde{P}) \in V(F') \\ P & \text{otw. } [\tilde{P} = P] \end{cases}$$

$$f(\tilde{v}, \tilde{P}) = (f_v(\tilde{v}, \tilde{P}), f_p(\tilde{v}, \tilde{P}))$$

$$f_B(\tilde{\zeta}) = b(f(\tilde{\mathcal{J}}(\tilde{\zeta}), \tilde{F}(\tilde{\zeta})))$$

We define our block sequence $\tilde{B}$ as $\tilde{B}(\tilde{\zeta}) = B(f_B(\tilde{\zeta}))$.

__Claim:__ $\tilde{\alpha}_{\tilde{p}}((\tilde{u}, \tilde{v}), \tilde{B}_i) = \alpha_{f_p(\tilde{u}, \tilde{P})}((f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})), B_i)$

__Claim:__ $(\tilde{u}, \tilde{v}) \in \tilde{E}(\tilde{P}^\circ)$ iff $(f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})) \in E(f_p(\tilde{u}, \tilde{P}))$

__Proof by cases $(\tilde{u}, \tilde{v})$.__

__Case $(\tilde{u}, \tilde{v}) = (\tilde{u}_{po}, t)$.__ Hence, by definition of f_v, $(f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})) = (u_{po}, t)$. Now
$(\tilde{u}, \tilde{v}) \in \tilde{E}(\tilde{P}^\circ) \overset{\text{def.}}{\text{iff}} \tilde{P} \neq \tilde{F}$ and $(u_{po}, t) \in E(P^\circ) \overset{\text{def.}}{\text{iff}} (f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})) \in E(f_p(\tilde{u}, \tilde{P}))$.

Where does $\tilde{P}$ come from? Where does $\tilde{x}$ come from?

__Claim:__ $\tilde{c}(\tilde{u}, \tilde{v}) = c(f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P}))$

__Claim:__ $\tilde{d}_{\tilde{p}} = d_{f_p(\tilde{u}, \tilde{P})}$

__Claim:__ $b(f(\tilde{\mathcal{J}}(\tilde{\zeta}(\tilde{u}, \tilde{P})), \tilde{P}) = b(f_v(\tilde{u}, \tilde{P}), f_p(\tilde{u}, \tilde{P}))$

$$\sum_{\tilde{p} \in \tilde{\mathcal{P}} : \tilde{\alpha}_{\tilde{p}}((\tilde{u}, \tilde{v}), \tilde{B}_{i-1}) = \text{active or } \tilde{\alpha}_{\tilde{p}}((\tilde{u}, \tilde{v}), \tilde{B}_i) = \text{active}} \tilde{d}_{\tilde{p}} \overset{\text{Claim} \sim}{=}$$

$$\sum_{\tilde{p} \in \tilde{\mathcal{P}} : \alpha_{f_p(\tilde{u}, \tilde{P})}((f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})), B_{i-1}) = \text{active or } \alpha_{f_p(\tilde{u}, \tilde{P})}((f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})), B_i) = \text{active}} \tilde{d}_{\tilde{p}} \overset{\text{Claim} \sim}{=}$$

$$d_{f_p(\tilde{u}, \tilde{P})} \leq$$

What does $\tilde{P}$ mean now?

$$\sum_{p \in \mathcal{P} : \alpha_p((f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})), B_{i-1}) = \text{active or } \alpha_p((f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})), B_i) = \text{active}} d_p \overset{\text{assumption}}{\leq} c(f_v(\tilde{u}, \tilde{P}), f_v(\tilde{v}, \tilde{P})) \overset{\text{Claim} \sim}{=} \tilde{c}(\tilde{u}, \tilde{v})$$