

September 4, 2013 – ESA, Sophia Antipolis

The Online Replacement Path Problem

Marco Senatore, University of Rome “Tor Vergata”

With David Adjiashvili (ETH) and Gianpaolo Oriolo (University of Rome “Tor Vergata”)

Introduction

Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .

Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.

Introduction

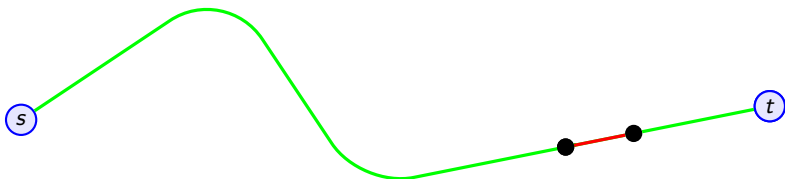
- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.

Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .

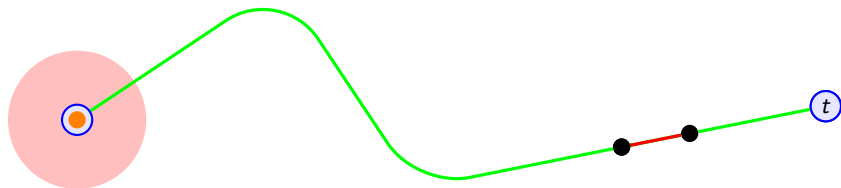
Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



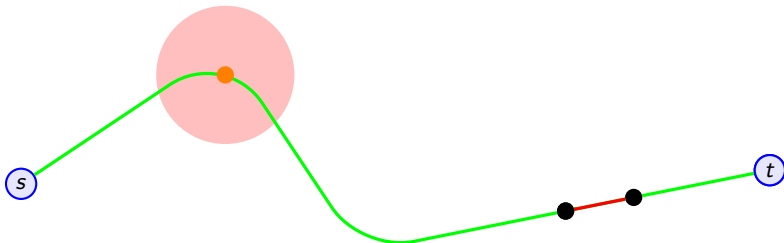
Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



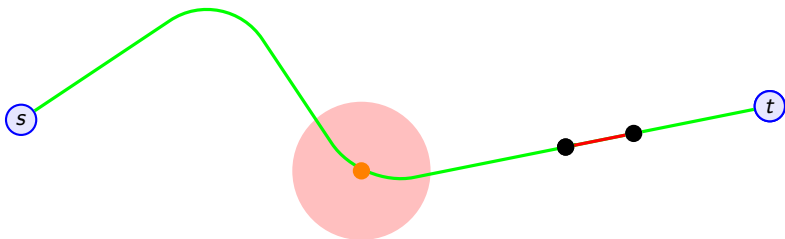
Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



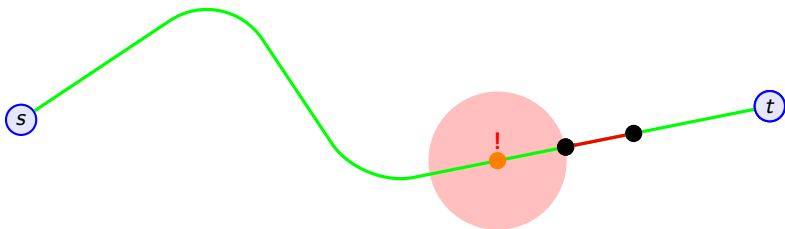
Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



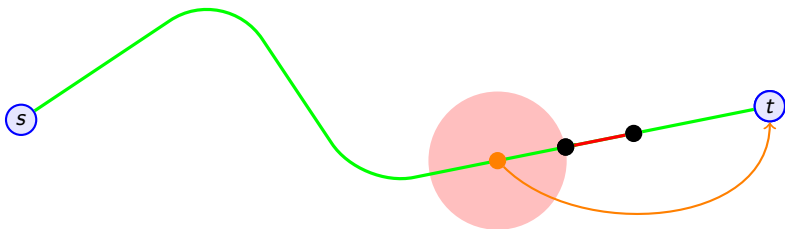
Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



Introduction

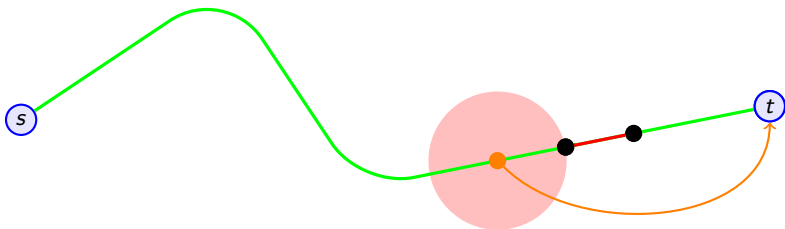
- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



- As soon as the RM discovers the failure, it is allowed to take a **detour** to reach t .

Introduction

- Given an edge-weighted graph $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$, a destination $t \in V$ and an s - t path P .
- Assume at most one edge $e \in E$ can fail.
- Routing Mechanism (RM) wants to route a package along P without knowing the identity of the failed edge.
- RM can discover the identity of e while traversing P .



- As soon as the RM discovers the failure, it is allowed to take a **detour** to reach t .
- **Goal:** Provide an s - t path to the RM, such that the **worst-case** total travel time is minimized.

Introduction

Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.

Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.

Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:



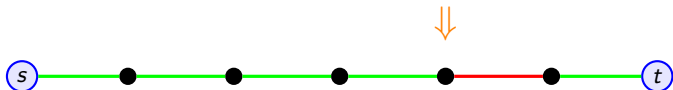
Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it → **Online Replacement Path (ORP)**;



Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it → **Online Replacement Path (ORP)**;



Introduction

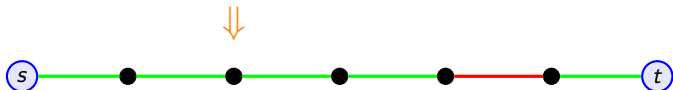
- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it → **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e on P → **k -Hop ORP**;



Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it \rightarrow **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e **on** $P \rightarrow$ **k -Hop ORP**;

$k = 2$



Introduction

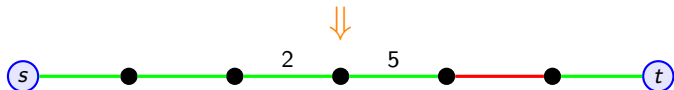
- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it \rightarrow **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e on $P \rightarrow$ **k -Hop ORP**;
 - 3 it reaches a node that is at distance at most R from the closer endpoint of e on $P \rightarrow$ **Radius ORP**;



Introduction

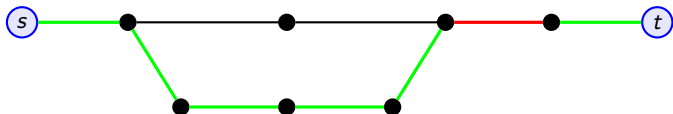
- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it \rightarrow **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e **on P** \rightarrow **k -Hop ORP**;
 - 3 it reaches a node that is at distance at most R from the closer endpoint of e **on P** \rightarrow **Radius ORP**;

$$R = 6$$



Introduction

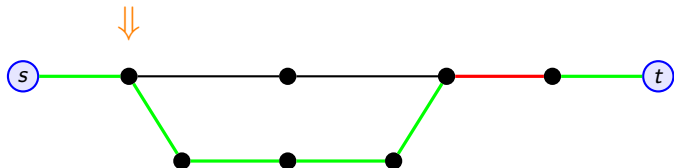
- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it \rightarrow **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e **on P** \rightarrow **k -Hop ORP**;
 - 3 it reaches a node that is at distance at most R from the closer endpoint of e **on P** \rightarrow **Radius ORP**;
 - 4 it reaches a node of P that is at most k hops away from e **on G** \rightarrow **Strong k -Hop ORP**



Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it → **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e on P → **k -Hop ORP**;
 - 3 it reaches a node that is at distance at most R from the closer endpoint of e on P → **Radius ORP**;
 - 4 it reaches a node of P that is at most k hops away from e on G → **Strong k -Hop ORP**

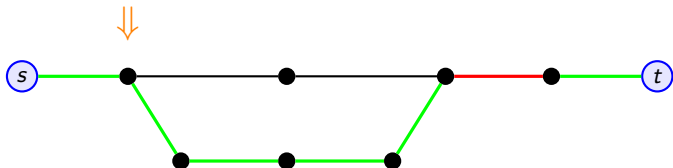
$$k = 2$$



Introduction

- RM wants to route a package along P without knowing the identity of the failed edge.
- Different variants of the problem can be defined.
- The RM discovers the identity of the failed edge e as soon as:
 - 1 it tries to cross it → **Online Replacement Path (ORP)**;
 - 2 it reaches a node that is at most k hops away from e **on P** → **k -Hop ORP**;
 - 3 it reaches a node that is at distance at most R from the closer endpoint of e **on P** → **Radius ORP**;
 - 4 it reaches a node of P that is at most k hops away from e **on G** → **Strong k -Hop ORP**

$$k = 2$$



- As soon as RM is aware of the failed edge, it follows the shortest path in the residual graph.

Some Notations

Some Notations

- For each $v \in V$,
 - 1 $\mathcal{P}_{v,t}$ is the set of v - t paths;
 - 2 $\pi(v) :=$ shortest v - t path length.
- For $e \in E$ and $u \in V$:
 - 1 let Q_u^{-e} be some fixed shortest u - t path in $G \setminus e$;
 - 2 $\pi^{-e}(u) = \ell(Q_u^{-e})$.

The Online Replacement Path Problem

The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

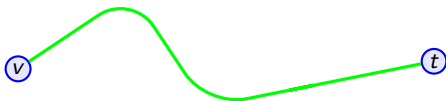
- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.

The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.



The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.

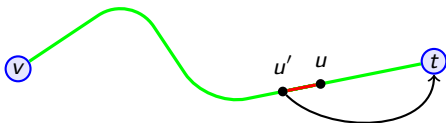


The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.

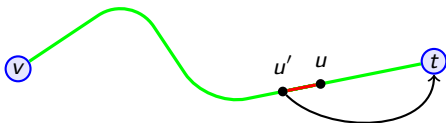


The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.



Definition (Robust length)

The **robust length** of the v - t path P is

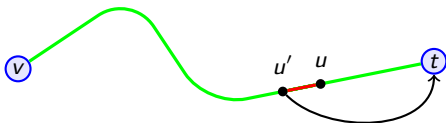
$$\text{Val}(P) = \max_{e \in E} \ell(P^{-e}).$$

The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.



Definition (Robust length)

The **robust length** of the v - t path P is

$$\text{Val}(P) = \max_{e \in E} \ell(P^{-e}).$$

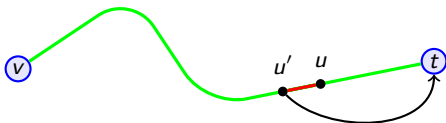
- For each $v \in V$, we define $\pi_{rob}(v)$ as the minimum of $\text{Val}(P)$ over all $P \in \mathcal{P}_{v,t}$.

The Online Replacement Path Problem

Definition (Detour)

Given a path $P \in \mathcal{P}_{v,t}$ and an edge $e = uu' \in E$, the **detour** P^{-e} is:

- P if $uu' \notin P$;
- the walk $P[v, u] \oplus Q_u^{-uu'}$ if $uu' \in P$.



Definition (Robust length)

The **robust length** of the v - t path P is

$$\text{Val}(P) = \max_{e \in E} \ell(P^{-e}).$$

- For each $v \in V$, we define $\pi_{rob}(v)$ as the minimum of $\text{Val}(P)$ over all $P \in \mathcal{P}_{v,t}$.
- **Find:** an s - t path with minimum robust length.

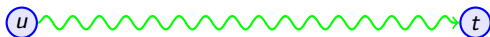
The Online Replacement Path Problem

The Online Replacement Path Problem

- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .

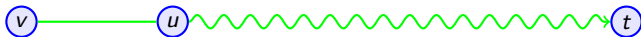
The Online Replacement Path Problem

- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .
- Consider P_u .



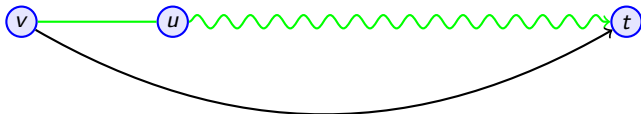
The Online Replacement Path Problem

- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .
- Consider P_u .



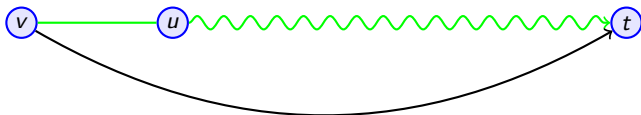
The Online Replacement Path Problem

- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .
- Consider P_u . Then $\text{Val}(vu \oplus P_u) = \max\{\ell(vu) + \text{Val}(P_u), \pi^{-vu}(v)\}$.



The Online Replacement Path Problem

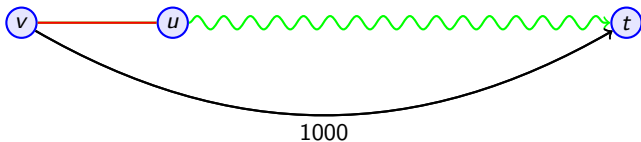
- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .
- Consider P_u . Then $\text{Val}(vu \oplus P_u) = \max\{\ell(vu) + \text{Val}(P_u), \pi^{-vu}(v)\}$.



- **[Monotonicity]:** $\text{Val}(P) \geq \text{Val}(P[x, t])$, whenever $x \in V(P)$.

The Online Replacement Path Problem

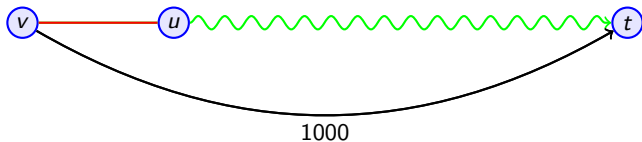
- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .
- Consider P_u . Then $\text{Val}(vu \oplus P_u) = \max\{\ell(vu) + \text{Val}(P_u), \pi^{-vu}(v)\}$.



- **[Monotonicity]:** $\text{Val}(P) \geq \text{Val}(P[x, t])$, whenever $x \in V(P)$.

The Online Replacement Path Problem

- $\text{Val}(P)$ is finite iff every $v \in P$ is 2-connected to t .
- Consider P_u . Then $\text{Val}(vu \oplus P_u) = \max\{\ell(vu) + \text{Val}(P_u), \pi^{-vu}(v)\}$.



- **[Monotonicity]:** $\text{Val}(P) \geq \text{Val}(P[x, t])$, whenever $x \in V(P)$.

Lemma (Weak optimality principle)

Let $P_v \in \mathcal{P}_{v,t}$ be an optimal path from v , $u \in V(P_v)$ and $P_u \in \mathcal{P}_{u,t}$ be an optimal path from u . Then the path $P'_v = P_v[v, u] \oplus P_u$ satisfies $\text{Val}^k(P'_v) = \text{Val}^k(P_v)$, namely it is also optimal from v .

The Online Replacement Path Problem

The Online Replacement Path Problem

- We devise a label setting algorithm.

The Online Replacement Path Problem

- We devise a label setting algorithm.

Lemma

Let $U \subset V$, with $t \in U$, be a set of nodes for which π_{rob} is known, and let $vu \in \delta(U)$:

$$vu = \arg \min_{zw \in E: w \in U, z \notin U} \max\{\ell(zw) + \pi_{rob}(w), \pi^{-zw}(z)\}.$$

Then $\pi_{rob}(v) = \text{Val}(vu \oplus P_u)$, where P_u is any optimal (robust) u - t path .

The Online Replacement Path Problem

- We devise a label setting algorithm.

Lemma

Let $U \subset V$, with $t \in U$, be a set of nodes for which π_{rob} is known, and let $vu \in \delta(U)$:

$$vu = \arg \min_{zw \in E: w \in U, z \notin U} \max\{\ell(zw) + \pi_{rob}(w), \pi^{-zw}(z)\}.$$

Then $\pi_{rob}(v) = \text{Val}(vu \oplus P_u)$, where P_u is any optimal (robust) u - t path .

Theorem

Given an instance of ORP, the values π_{rob} and the corresponding paths can be computed in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.

The Online Replacement Path Problem

- We devise a label setting algorithm.

Lemma

Let $U \subset V$, with $t \in U$, be a set of nodes for which π_{rob} is known, and let $vu \in \delta(U)$:

$$vu = \arg \min_{zw \in E: w \in U, z \notin U} \max\{\ell(zw) + \pi_{rob}(w), \pi^{-zw}(z)\}.$$

Then $\pi_{rob}(v) = \text{Val}(vu \oplus P_u)$, where P_u is any optimal (robust) u - t path .

Theorem

Given an instance of ORP, the values π_{rob} and the corresponding paths can be computed in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.

- This complexity bound builds upon a result by Proietti et al. of '99 providing an $O(m\alpha(m, n))$ algorithm to compute $\pi^{-e}(u)$, $\forall e = uu' \in E$.

The Online Replacement Path Problem

- We devise a label setting algorithm.

Lemma

Let $U \subset V$, with $t \in U$, be a set of nodes for which π_{rob} is known, and let $vu \in \delta(U)$:

$$vu = \arg \min_{zw \in E: w \in U, z \notin U} \max\{\ell(zw) + \pi_{rob}(w), \pi^{-zw}(z)\}.$$

Then $\pi_{rob}(v) = \text{Val}(vu \oplus P_u)$, where P_u is any optimal (robust) u - t path .

Theorem

Given an instance of ORP, the values π_{rob} and the corresponding paths can be computed in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.

- This complexity bound builds upon a result by Proietti et al. of '99 providing an $O(m\alpha(m, n))$ algorithm to compute $\pi^{-e}(u)$, $\forall e = uu' \in E$.
- The same complexity bound is attained in Bar-Noy and Schieber '91, "The Canadian Traveller Problem".

Robust Length vs. Nominal Length

Robust Length vs. Nominal Length

- In some applications faults occur rarely, hence it is preferred to have the cost of the chosen path as low as possible.

Robust Length vs. Nominal Length

- In some applications faults occur rarely, hence it is preferred to have the cost of the chosen path as low as possible.
- At the same time we could have a budget B on the robust length.

Robust Length vs. Nominal Length

- In some applications faults occur rarely, hence it is preferred to have the cost of the chosen path as low as possible.
- At the same time we could have a budget B on the robust length.
- We aim at solving:

$$P^* := \arg \min_{P \in \mathcal{P}_{s,t}: \text{Val}(P) \leq B} \ell(P).$$

Robust Length vs. Nominal Length

- In some applications faults occur rarely, hence it is preferred to have the cost of the chosen path as low as possible.
- At the same time we could have a budget B on the robust length.
- We aim at solving:

$$P^* := \arg \min_{P \in \mathcal{P}_{s,t}: \text{Val}(P) \leq B} \ell(P).$$

- We can solve this problem with a simple modification of Dijkstra's algorithm.

Robust Length vs. Nominal Length

- In some applications faults occur rarely, hence it is preferred to have the cost of the chosen path as low as possible.
- At the same time we could have a budget B on the robust length.
- We aim at solving:

$$P^* := \arg \min_{P \in \mathcal{P}_{s,t}: \text{Val}(P) \leq B} \ell(P).$$

- We can solve this problem with a simple modification of Dijkstra's algorithm.

Algorithm

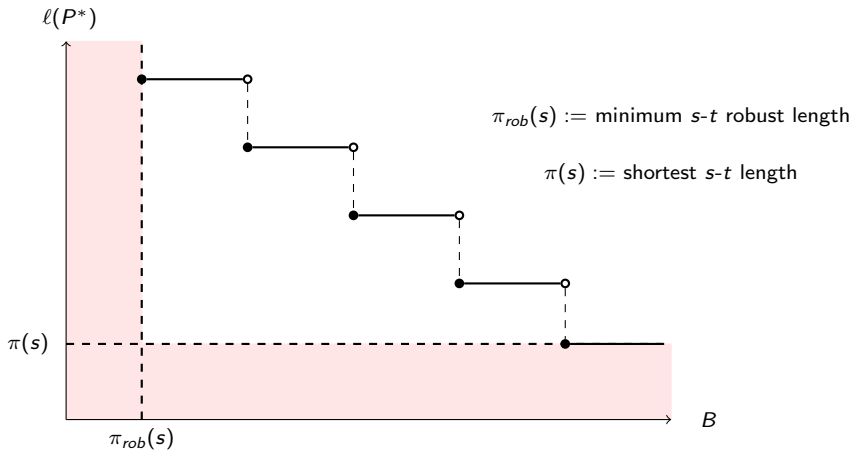
```

1:  $S = \emptyset; \bar{S} = V \quad d(s) = 0; \quad d(u) = \infty \forall u \in V \setminus s$ 
2: while  $t \notin S$  do
3:   Find  $u = \arg \min_{z \in \bar{S}} d(z)$ 
4:    $S = S + u \quad \bar{S} = \bar{S} - u$ 
5:   for  $q \in N(u) \setminus S$  do
6:     if  $d(u) + \ell(uq) \leq d(q)$  and  $d(u) + \pi^{-uq}(u) \leq B$  then
7:        $d(q) = d(u) + \ell(uq)$ 

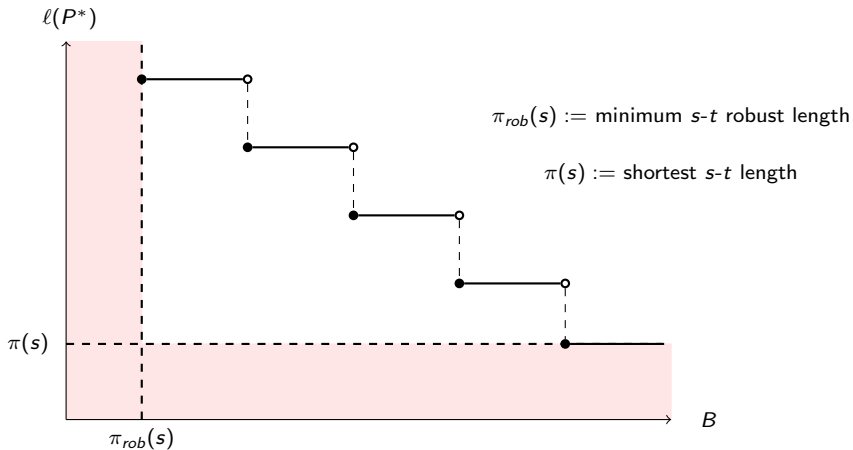
```


Robust Length vs. Nominal Length

Robust Length vs. Nominal Length

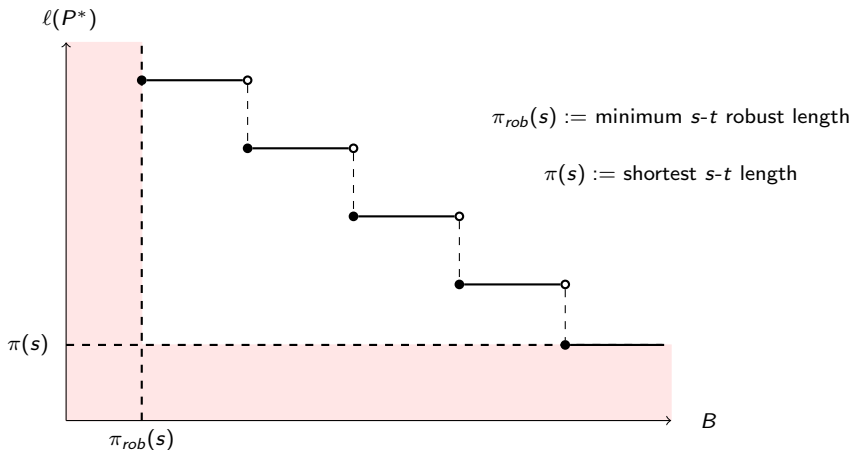


Robust Length vs. Nominal Length



- The number of steps is $O(m)$.

Robust Length vs. Nominal Length



- The number of steps is $O(m)$.
- The profile can be found in $O(m * (m + n \log n))$.

k -Hop ORP

k -Hop ORP

- The RM is informed about the failure of an edge $e \in E$ as soon as it reaches a node that is at most k hops away from e **on** P .

k -Hop ORP

- The RM is informed about the failure of an edge $e \in E$ as soon as it reaches a node that is at most k hops away from e on P .

 $k = 2$ 

k-Hop ORP

k-Hop ORP

- Consider a path P and an edge $e \in E(P)$. Denote by $v(P, e)$ the first node u of P such that e is **visible** from u .

k -Hop ORP

- Consider a path P and an edge $e \in E(P)$. Denote by $v(P, e)$ the first node u of P such that e is **visible** from u .

Definition (Detour)

Given a path $P \in \mathcal{P}_{u,t}$ and an edge $e \in E$, the detour P^{-e} is:

- the walk $P[u, v(P, e)] \oplus Q_{v(P, e)}^{-e}$ if $e \in P$;
- P otherwise.

k -Hop ORP

- Consider a path P and an edge $e \in E(P)$. Denote by $v(P, e)$ the first node u of P such that e is **visible** from u .

Definition (Detour)

Given a path $P \in \mathcal{P}_{u,t}$ and an edge $e \in E$, the detour P^{-e} is:

- the walk $P[u, v(P, e)] \oplus Q_{v(P, e)}^{-e}$ if $e \in P$;
- P otherwise.

Definition (k -Robust length)

The **k -robust length** of the u - t path P is

$$\text{Val}^k(P) = \max_{e \in E} \ell(P^{-e}).$$

k -Hop ORP

- Consider a path P and an edge $e \in E(P)$. Denote by $v(P, e)$ the first node u of P such that e is **visible** from u .

Definition (Detour)

Given a path $P \in \mathcal{P}_{u,t}$ and an edge $e \in E$, the detour P^{-e} is:

- the walk $P[u, v(P, e)] \oplus Q_{v(P, e)}^{-e}$ if $e \in P$;
- P otherwise.

Definition (k -Robust length)

The **k -robust length** of the u - t path P is

$$\text{Val}^k(P) = \max_{e \in E} \ell(P^{-e}).$$

- **Find:** an s - t path with minimum k -robust length.

k -Hop ORP

- Consider a path P and an edge $e \in E(P)$. Denote by $v(P, e)$ the first node u of P such that e is **visible** from u .

Definition (Detour)

Given a path $P \in \mathcal{P}_{u,t}$ and an edge $e \in E$, the detour P^{-e} is:

- the walk $P[u, v(P, e)] \oplus Q_{v(P, e)}^{-e}$ if $e \in P$;
- P otherwise.

Definition (k -Robust length)

The **k -robust length** of the u - t path P is

$$\text{Val}^k(P) = \max_{e \in E} \ell(P^{-e}).$$

- **Find:** an s - t path with minimum k -robust length.

k-Hop ORP

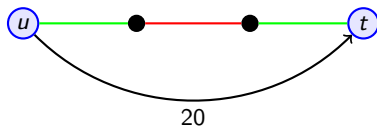
k-Hop ORP

- More challenging than ORP: the *k*-robust length is not monotonic!

k-Hop ORP

- More challenging than ORP: the *k*-robust length is not monotonic!

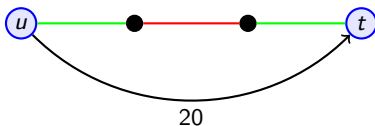
$$k = 2$$



k-Hop ORP

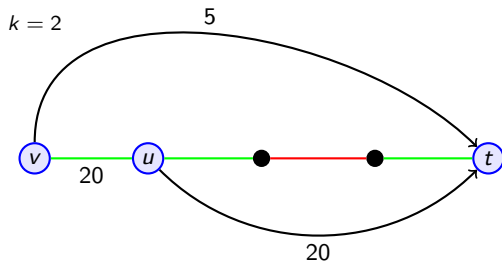
- More challenging than ORP: the *k*-robust length is not monotonic!

$$k = 2$$



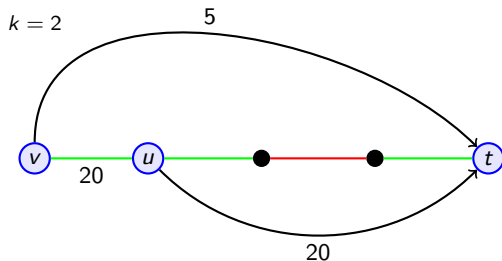
k-Hop ORP

- More challenging than ORP: the *k*-robust length is not monotonic!



k -Hop ORP

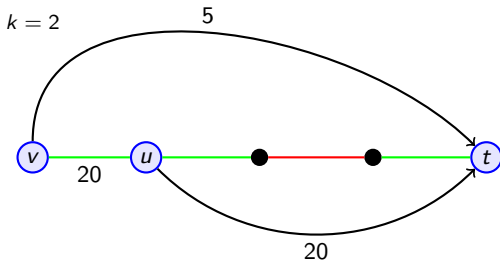
- More challenging than ORP: the k -robust length is not monotonic!



- Property:* If $P \in \mathcal{P}_{u,t}$ is such that $\text{Val}^k(P) = \pi^{-e}(u)$ for some $e \in P$ visible from u , then P is an optimal path from u .

k-Hop ORP

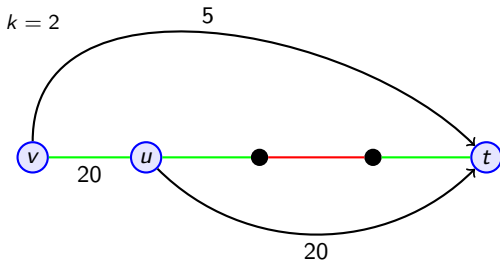
- More challenging than ORP: the *k*-robust length is not monotonic!



- Property:* If $P \in \mathcal{P}_{u,t}$ is such that $\text{Val}^k(P) = \pi^{-e}(u)$ for some $e \in P$ visible from u , then P is an optimal path from u . \implies **Preprocessing step!**

k -Hop ORP

- More challenging than ORP: the k -robust length is not monotonic!



- Property:* If $P \in \mathcal{P}_{u,t}$ is such that $\text{Val}^k(P) = \pi^{-e}(u)$ for some $e \in P$ visible from u , then P is an optimal path from u . \implies **Preprocessing step!**
- Weak optimality principle** still holds for k -Hop ORP

k-Hop ORP and Radius ORP

k-Hop ORP and Radius ORP

Theorem

*Given an instance of *k*-Hop ORP we can compute the *v*-*t* path with minimum *k*-robust length for any $v \in V$ in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.*

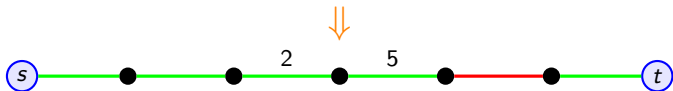
k-Hop ORP and Radius ORP

Theorem

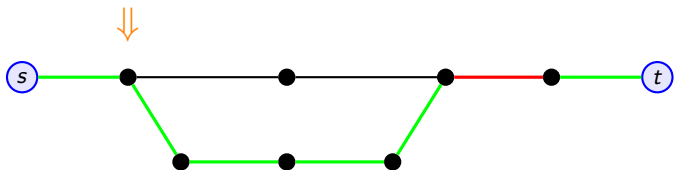
Given an instance of *k*-Hop ORP we can compute the *v*-*t* path with minimum *k*-robust length for any $v \in V$ in time $O(m + n \log n)$ in undirected graphs, and $O(mn + n^2 \log n)$ in directed graphs.

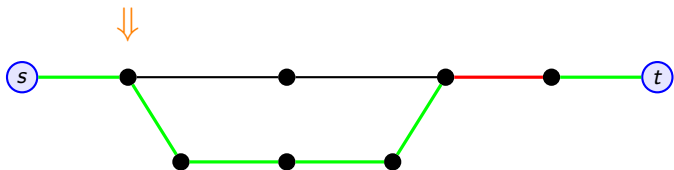
- Our algorithm for *k*-Hop ORP solves Radius ORP as well.

$$R = 6$$



Strong k -Hop ORP

Strong k -Hop ORP $k = 2$ 

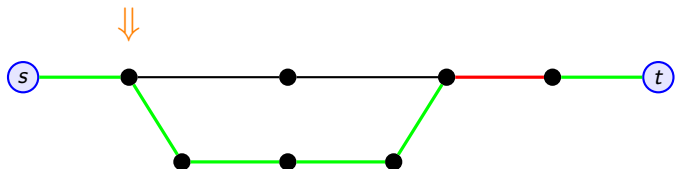
Strong k -Hop ORP $k = 2$ 

Theorem

For any $\epsilon > 0$ it is NP-hard to approximate Strong 1-Hop ORP within a factor of $3 - \epsilon$ in undirected graphs. In directed graphs it is strongly NP-hard to decide if there exists a path with finite robust length.

Strong k -Hop ORP

$k = 2$



Theorem

For any $\epsilon > 0$ it is NP-hard to approximate Strong 1-Hop ORP within a factor of $3 - \epsilon$ in undirected graphs. In directed graphs it is strongly NP-hard to decide if there exists a path with finite robust length.

- Any shortest path is a 3 approximation for undirected graphs.

A Game Theory Perspective

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:
 - 1 The **path builder** wants to route a package from s to t as quickly as possible.

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:
 - 1 The **path builder** wants to route a package from s to t as quickly as possible.
 - 2 The **interdictor** wants to increase the s - t shortest distance as much as possible by removing an edge from the network.

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:
 - 1 The **path builder** wants to route a package from s to t as quickly as possible.
 - 2 The **interdictor** wants to increase the s - t shortest distance as much as possible by removing an edge from the network.
- The strategies of the **interdictor** are edges $e \in E$; if she moves **first**, she needs to solve the **Most Vital Arc Problem**. Let $z^*(MVA)$ be the optimal value.

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:
 - 1 The **path builder** wants to route a package from s to t as quickly as possible.
 - 2 The **interdictor** wants to increase the s - t shortest distance as much as possible by removing an edge from the network.
- The strategies of the **interdictor** are edges $e \in E$; if she moves **first**, she needs to solve the **Most Vital Arc Problem**. Let $z^*(MVA)$ be the optimal value.
- The strategies of the **path builder** are s - t paths; if she moves **first**, she needs to solve the **ORP Problem**. Let $z^*(ORP)$ be the optimal value.

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:
 - 1 The **path builder** wants to route a package from s to t as quickly as possible.
 - 2 The **interdictor** wants to increase the s - t shortest distance as much as possible by removing an edge from the network.
- The strategies of the **interdictor** are edges $e \in E$; if she moves **first**, she needs to solve the **Most Vital Arc Problem**. Let $z^*(MVA)$ be the optimal value.
- The strategies of the **path builder** are s - t paths; if she moves **first**, she needs to solve the **ORP Problem**. Let $z^*(ORP)$ be the optimal value.
- We call the **ORP Game**, that game in which both the **path builder** and the **interdictor** communicate their strategies at the **same time**.

A Game Theory Perspective

- Given $G = (V, E, \ell)$, $\ell : E \rightarrow \mathbb{R}^+$, a source $s \in V$ and a destination $t \in V$.
- We consider two players, a **path builder** and an **interdictor**:
 - 1 The **path builder** wants to route a package from s to t as quickly as possible.
 - 2 The **interdictor** wants to increase the s - t shortest distance as much as possible by removing an edge from the network.
- The strategies of the **interdictor** are edges $e \in E$; if she moves **first**, she needs to solve the **Most Vital Arc Problem**. Let $z^*(MVA)$ be the optimal value.
- The strategies of the **path builder** are s - t paths; if she moves **first**, she needs to solve the **ORP Problem**. Let $z^*(ORP)$ be the optimal value.
- We call the **ORP Game**, that game in which both the **path builder** and the **interdictor** communicate their strategies at the **same time**.

ORP Game

ORP Game

Theorem

Let P and e be optimal solutions to the ORP and MVA instances on $G = (V, E)$. Then (P, e) is a pure NE of the ORP Game if and only if $\text{Val}(P) = \pi^{-e}(s)$. Moreover, in this case, $\text{Val}(P) = z^*(\text{ORP}) = z^*(\text{MVA}) = \pi^{-e}(s)$.

- We can compute a pure NE or certify that no pure NE exists in $O(m + n \log n)$ for undirected graphs, in $O(mn + n^2 \log n)$ for directed graphs.

ORP Game

Theorem

Let P and e be optimal solutions to the ORP and MVA instances on $G = (V, E)$. Then (P, e) is a pure NE of the ORP Game if and only if $\text{Val}(P) = \pi^{-e}(s)$. Moreover, in this case, $\text{Val}(P) = z^*(\text{ORP}) = z^*(\text{MVA}) = \pi^{-e}(s)$.

- We can compute a pure NE or certify that no pure NE exists in $O(m + n \log n)$ for undirected graphs, in $O(mn + n^2 \log n)$ for directed graphs.
- In the case where $z^*(\text{ORP}) \neq z^*(\text{MVA})$, the ORP Game will still admit a NE in mixed strategy.

ORP Game

Theorem

Let P and e be optimal solutions to the ORP and MVA instances on $G = (V, E)$. Then (P, e) is a pure NE of the ORP Game if and only if $\text{Val}(P) = \pi^{-e}(s)$. Moreover, in this case, $\text{Val}(P) = z^*(\text{ORP}) = z^*(\text{MVA}) = \pi^{-e}(s)$.

- We can compute a pure NE or certify that no pure NE exists in $O(m + n \log n)$ for undirected graphs, in $O(mn + n^2 \log n)$ for directed graphs.
- In the case where $z^*(\text{ORP}) \neq z^*(\text{MVA})$, the ORP Game will still admit a NE in mixed strategy.
- **Question:** Is it possible to find a mixed NE in polynomial time?

Stochastic ORP

Stochastic ORP

- In this variant of ORP:
 - 1 at most one edge can fail;
 - 2 for each $e \in E$, $p(e) \in [0, 1]$ is the probability that e will be the unique edge to fail;
 - 3 $p(\emptyset) \in [0, 1]$ is the probability that no edge fails;
 - 4 it holds $p(\emptyset) + \sum_{e \in E} p(e) = 1$.

Definition

Given a node $v \in V$, the *expected robust length* of the v - t path P is

$$\mathbb{E}\text{Val}(P) = (p(\emptyset) + \sum_{uu' \notin P} p(uu'))\ell(P) + \sum_{uu' \in P} \left[p(uu')(\ell(P[v, u]) + \pi^{-uu'}(u)) \right].$$

- Stochastic ORP is to find, for some $v \in V$, a path P minimizing $\mathbb{E}\text{Val}(P)$ over all paths $P \in \mathcal{P}_{v,t}$.
- Stochastic ORP is NP-hard by reduction from the Hamiltonian Path problem.
- **Open question:** Is it possible to find a mixed NE in polynomial time?

Future work

Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.

Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!

Future work

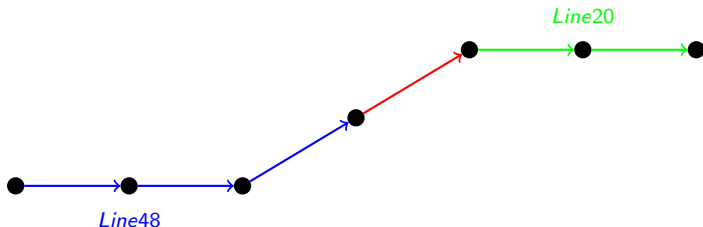
- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!
- On the other hand, customers want to take reliable journeys.

Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!
- On the other hand, customers want to take reliable journeys.
- We want to be robust with respect to *missing connections*.

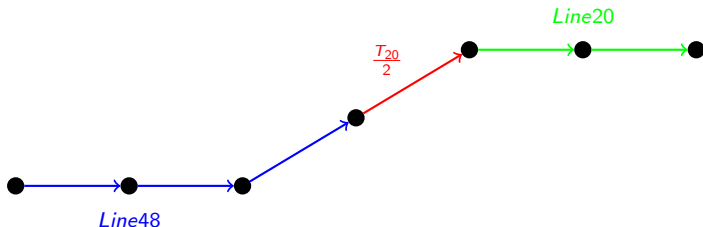
Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!
- On the other hand, customers want to take reliable journeys.
- We want to be robust with respect to *missing connections*.



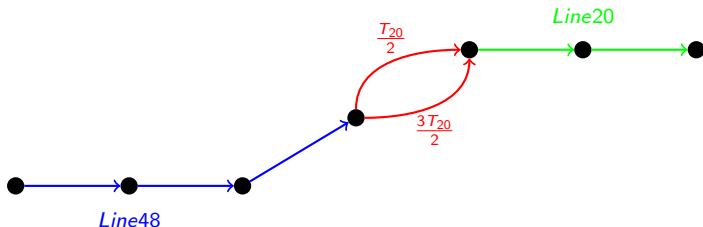
Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!
- On the other hand, customers want to take reliable journeys.
- We want to be robust with respect to *missing connections*.



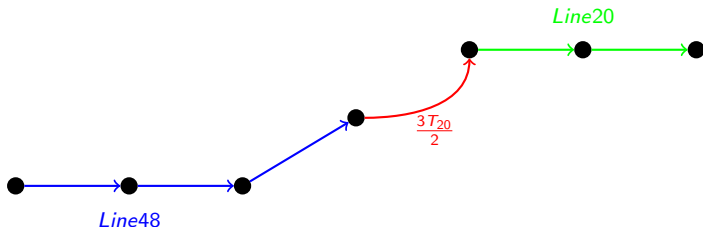
Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!
- On the other hand, customers want to take reliable journeys.
- We want to be robust with respect to *missing connections*.



Future work

- Collaboration with Prof. Italiano's group and the Public Transport (PT) Agency of Rome: historical and on-line data.
- PT in Rome is very unreliable!!
- On the other hand, customers want to take reliable journeys.
- We want to be robust with respect to *missing connections*.



Questions?

Thank You...