A Strongly Polynomial Algorithm for Finding a Shortest Non-zero Path in Group-Labeled Graphs

Yutaro Yamaguchi

Osaka University, Japan.

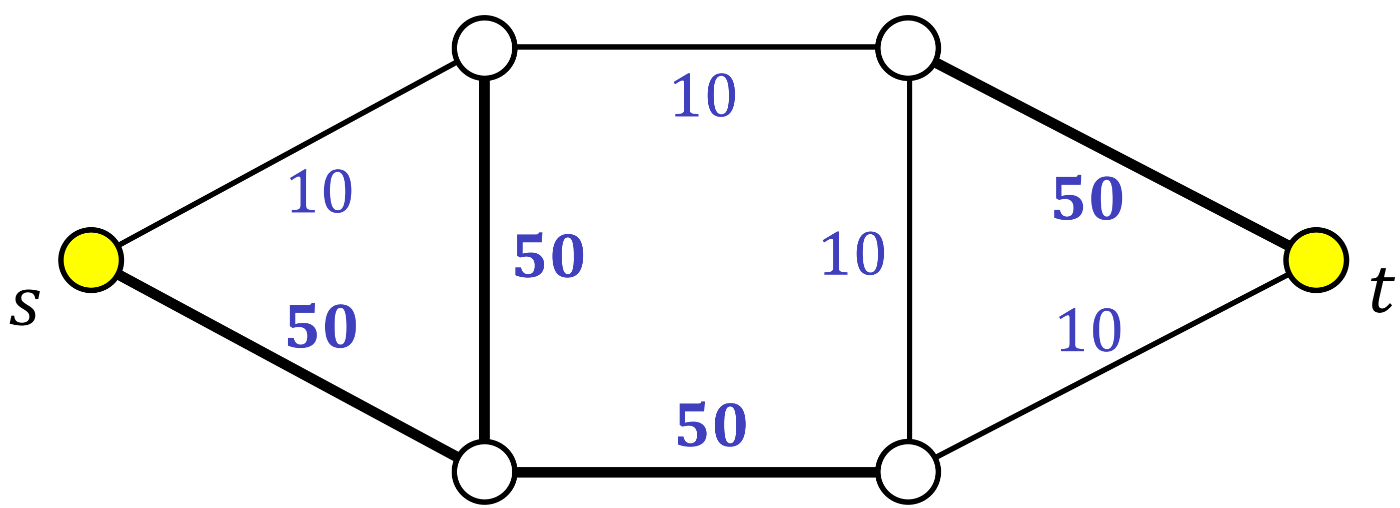
SODA'20 @Salt Lake City   Jan 7, 2020

Shortest Path Problem

Input $G = (V, E)$: Undirected Graph

$\ell \in \mathbf{R}_{\geq 0}^{E}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest s–t path P in G

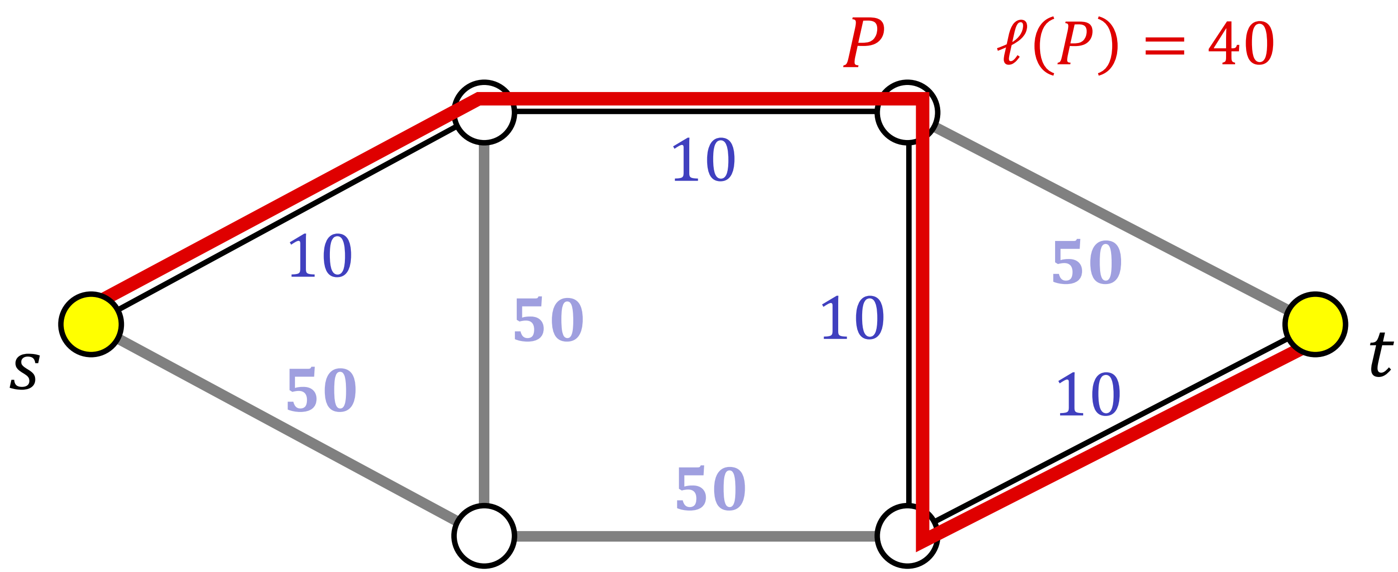


Shortest Path Problem

Input $G = (V, E)$: Undirected Graph

$\ell \in \mathbf{R}_{\geq 0}^{E}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest s–t path P in G



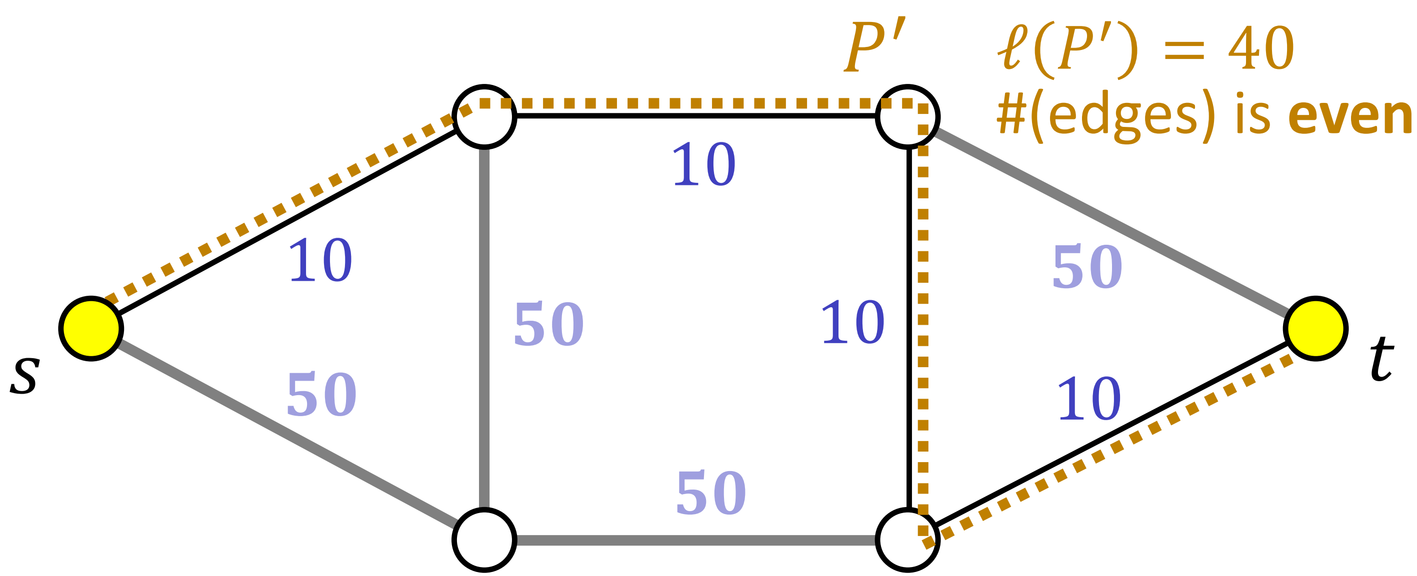
Solved by Dijkstra's Algorithm

Shortest Odd Path Problem

Input $G = (V, E)$: Undirected Graph

$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **odd** s–t path P in G

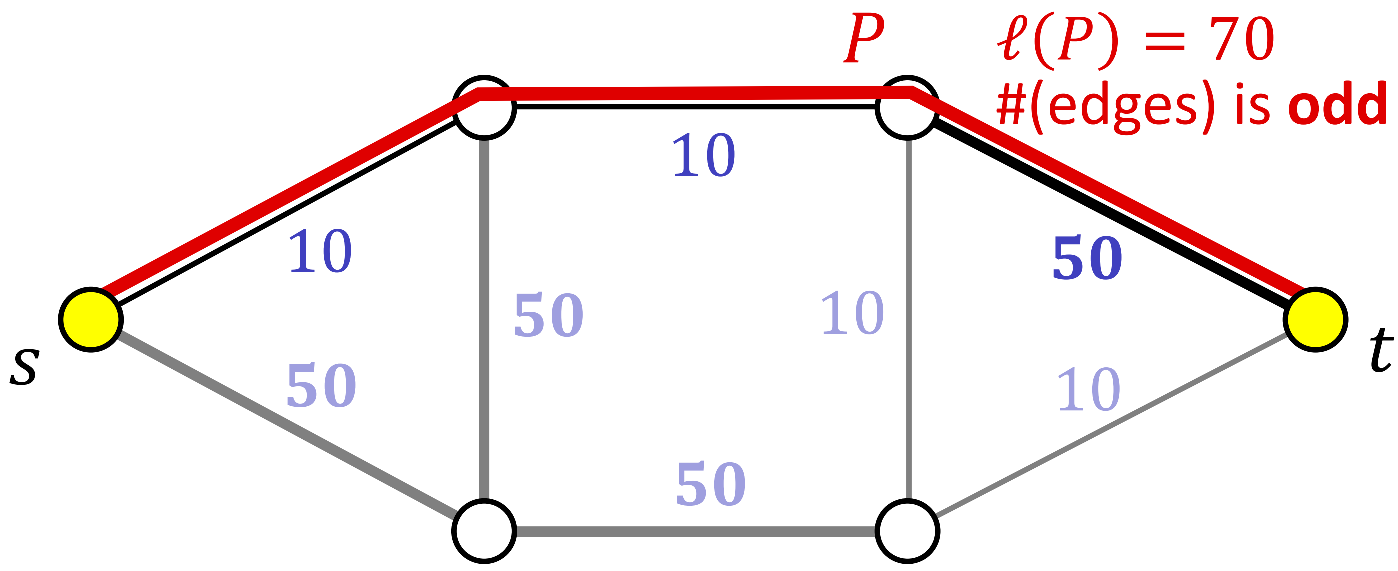


Shortest Odd Path Problem

Input $G = (V, E)$: Undirected Graph

$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **odd** s–t path P in G



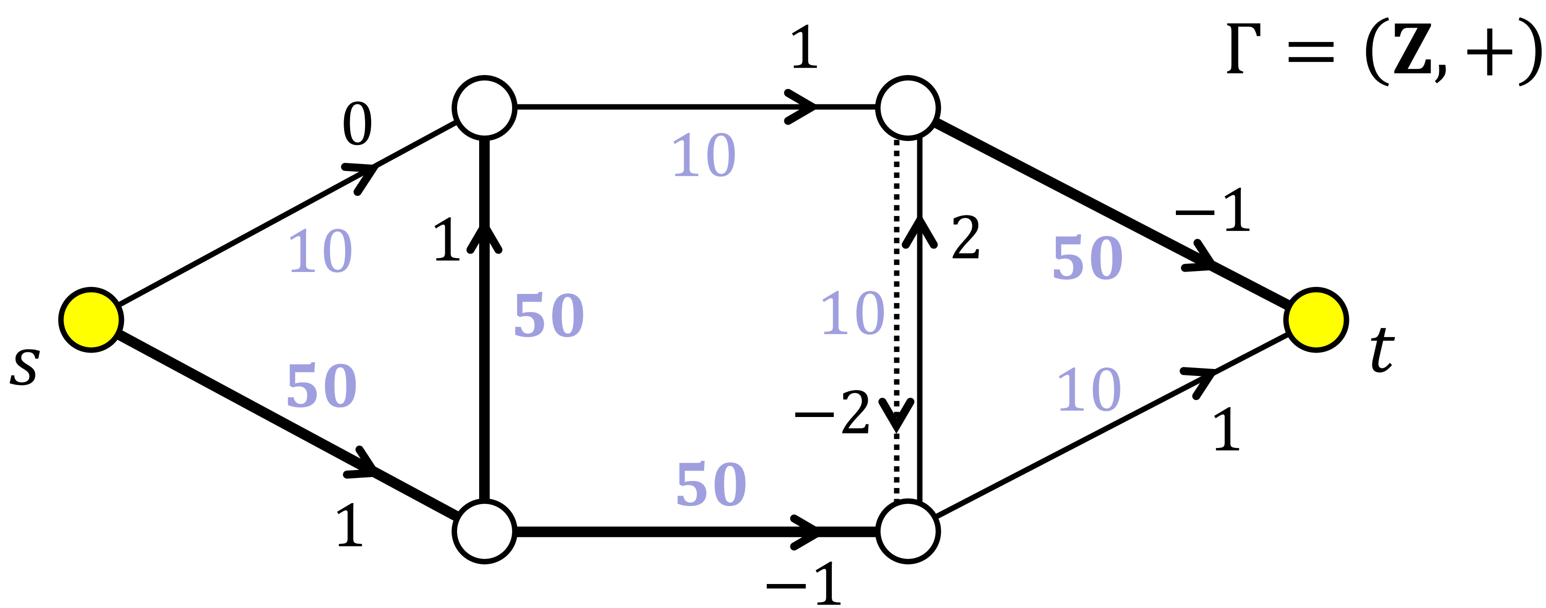
Solved via Weighted Matching

Shortest Non-zero Path Problem

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G

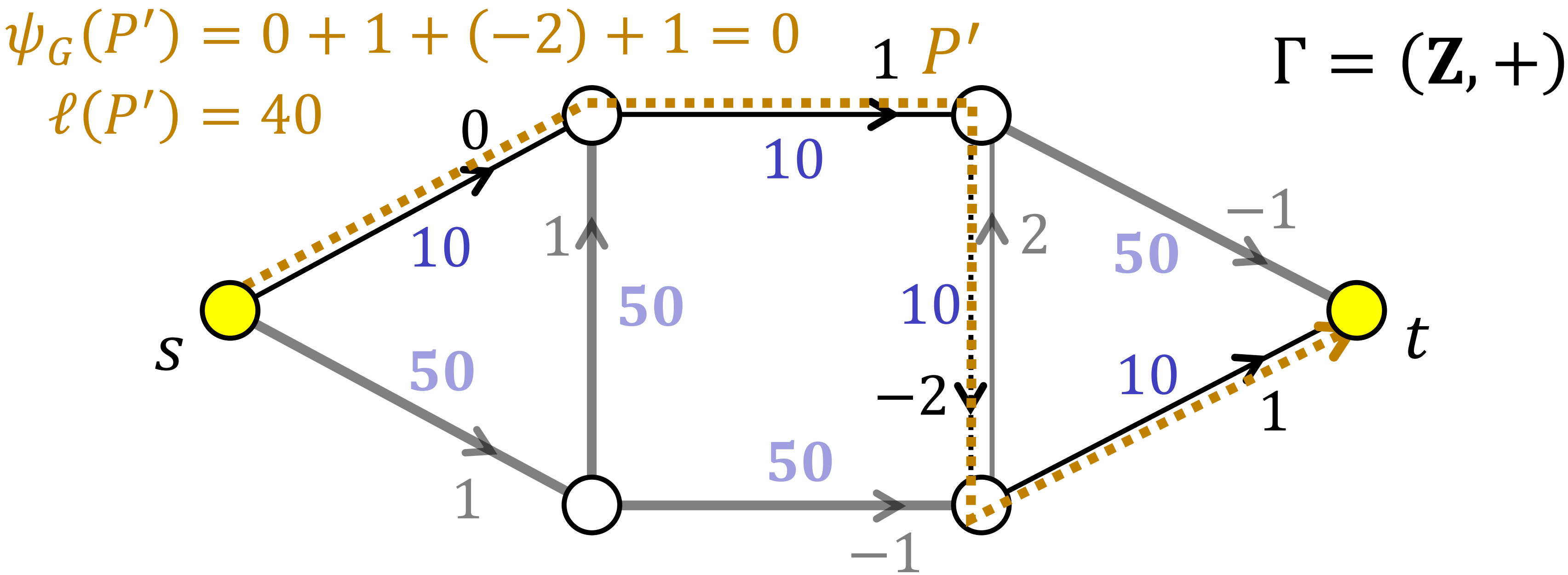


Shortest Non-zero Path Problem

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G

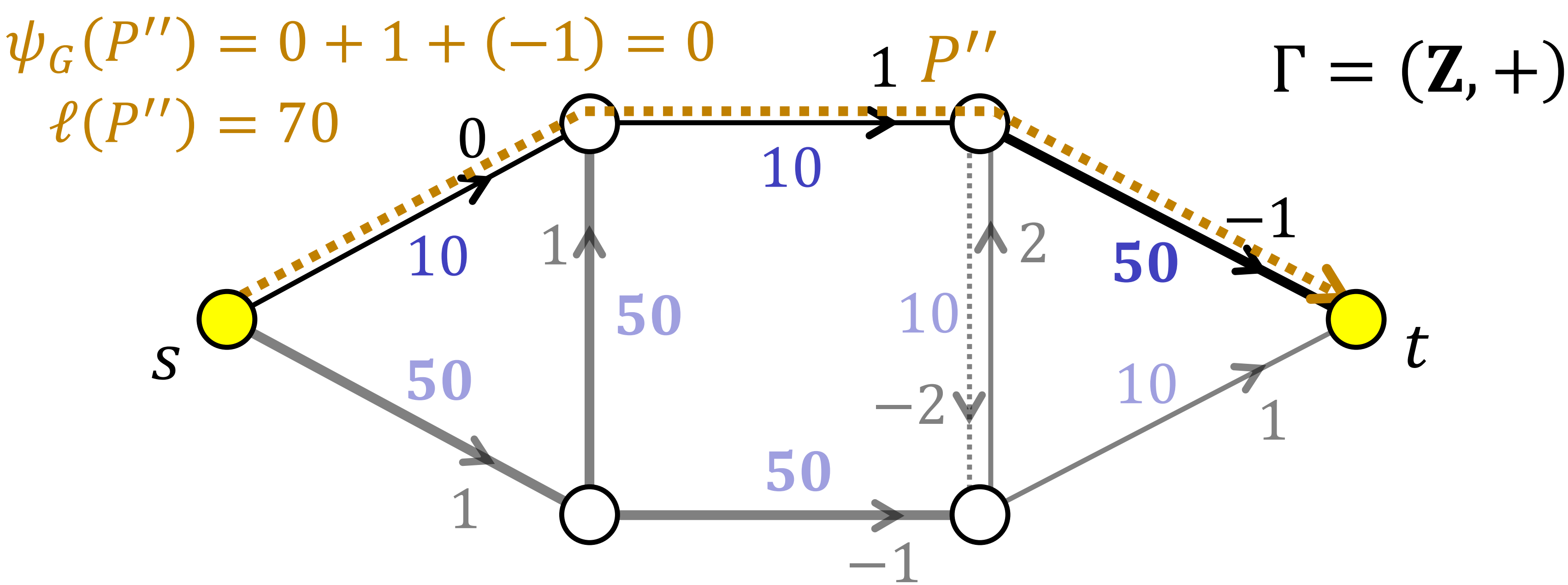


Shortest Non-zero Path Problem

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G



Shortest Non-zero Path Problem

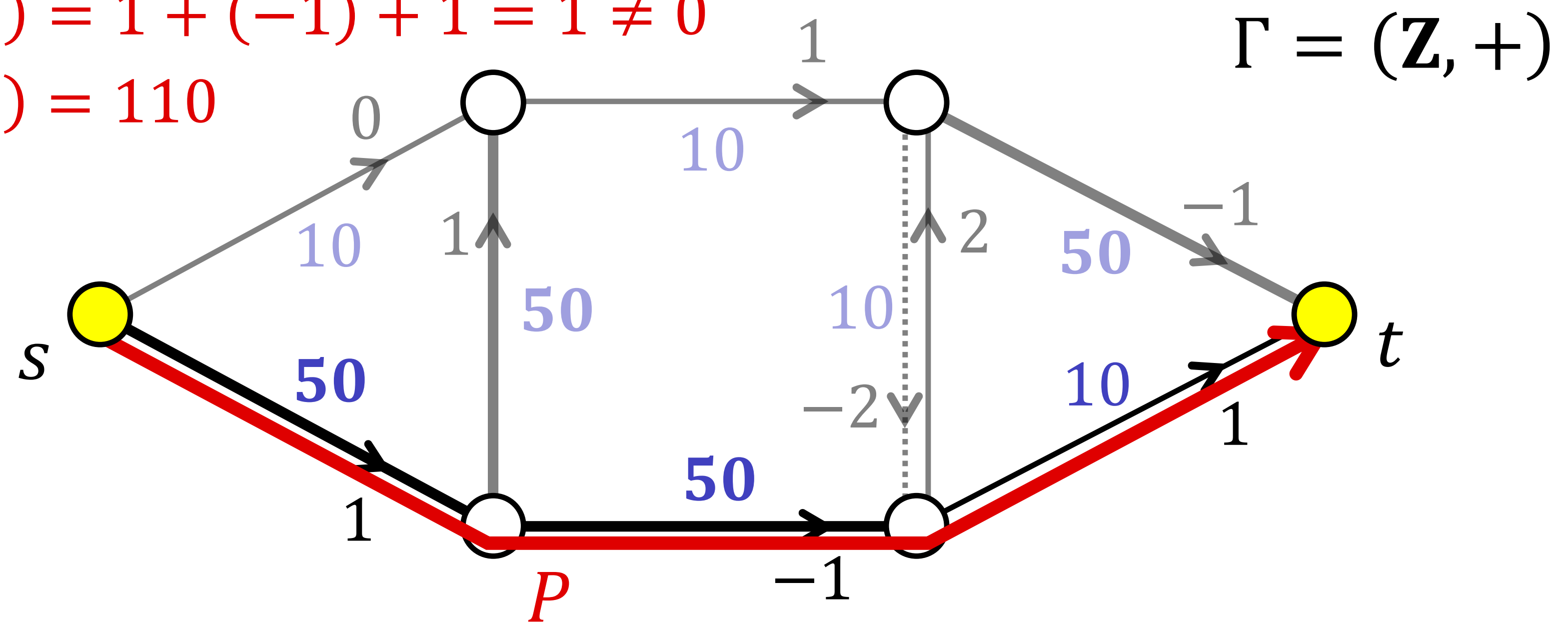
Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}^{E}_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G

$\psi_G(P) = 1 + (-1) + 1 = 1 \neq 0$

$\ell(P) = 110$



Shortest Non-zero Path Problem

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}_{\geq 0}^{E}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G

Thm. Solved by $\mathrm{O}(|V| \cdot |E|)$ Elementary Operations

[This Work]

- When $|\Gamma| = 2$, This Problem $\simeq$ **Shortest Odd Path Problem**
- When $\Gamma \simeq \mathbf{Z}_{n_1} \oplus \cdots \oplus \mathbf{Z}_{n_k}$ (i.e., Γ is finite & abelian),
  Randomized Pseudo-Poly via **Permanent Computation**
  [Kobayashi–Toyooka 2017]
- When $\Gamma \simeq \mathbf{Z}_{p_1} \oplus \cdots \oplus \mathbf{Z}_{p_k}$ (p_i: prime),
  Deterministic Strongly-Poly via **Weighted Linear Matroid Parity**
  [Y. 2016] + [Iwata–Kobayashi 2017]

Outline

- Algorithm Framework
  - Basic Idea
  - Auxiliary Problem (Shortest Unorthodox Path)
  - Main Lemma
- Key Structure: Lowest Blossoms
  - Detour yields a Shortest Unorthodox Path (SUP)
  - Shrinking preserves SUP Problem
- Conclusion

Outline

- Algorithm Framework
  - Basic Idea
  - Auxiliary Problem (Shortest Unorthodox Path)
  - Main Lemma
- Key Structure: Lowest Blossoms
  - Detour yields a Shortest Unorthodox Path (SUP)
  - Shrinking preserves SUP Problem
- Conclusion

Basic Idea

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)
$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

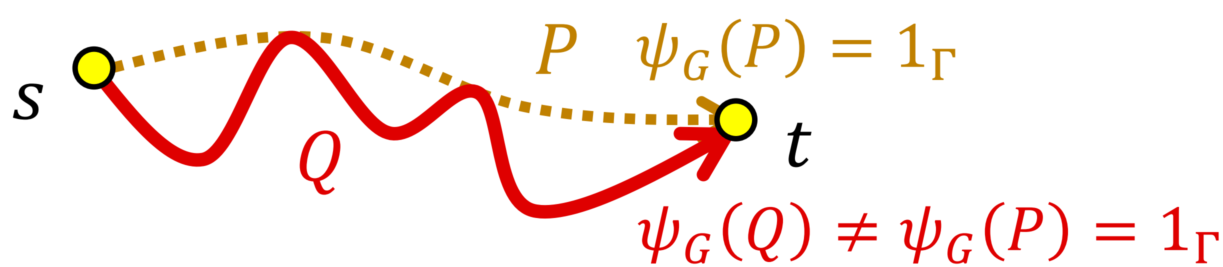
Goal Find a shortest **non-zero** s–t path P in G

1. Find a shortest s–t path P in G by Dijkstra's Algorithm
2. If P is non-zero ($\psi_G(P) \neq 1_\Gamma$), then return P
3. Otherwise, find and return an s–t path Q in G s.t. $\ell(Q)$ is minimized subject to $\psi_G(Q) \neq \psi_G(P)$

Basic Idea

Input	$G = (V, E)$: Γ-Labeled Graph (Γ: Group)
	$\ell \in \mathbf{R}_{\geq 0}^{E}$: Edge Length, $s, t \in V$: Terminals
Goal	Find a shortest **non-zero** s–t path P in G

1. Find a shortest s–t path P in G by Dijkstra's Algorithm
2. If P is non-zero ($\psi_G(P) \neq 1_\Gamma$), then return P
3. Otherwise, find and return an s–t path Q in G s.t. $\ell(Q)$ is minimized subject to $\psi_G(Q) \neq \psi_G(P)$



Auxiliary Problem for Main Task

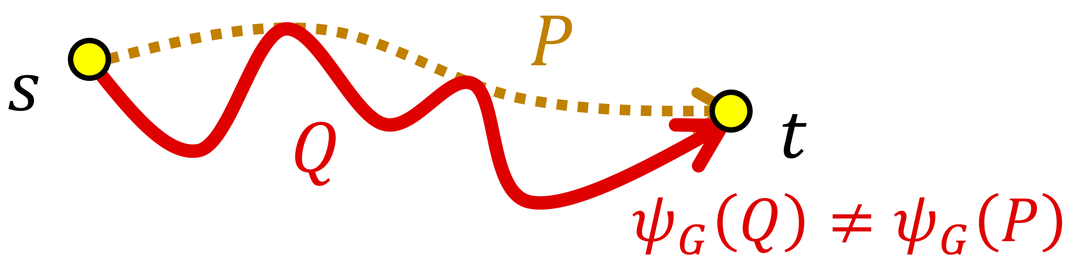
Input $(G=(V,E),\ell,s,t)$: Original Input

P: Shortest s–t Path in G

Goal Find a shortest **unorthodox** s–t path Q in G

1. Find a shortest s–t path P in G by Dijkstra's Algorithm
2. If P is non-zero ($\psi_G(P) \neq 1_\Gamma$), then return P
3. Otherwise, find and return an s–t path Q in G s.t.

$\ell(Q)$ is minimized subject to $\psi_G(Q) \neq \psi_G(P)$



Auxiliary Problem for Main Task

Input $(G = (V, E), \ell, s, t)$: Original Input

$T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s

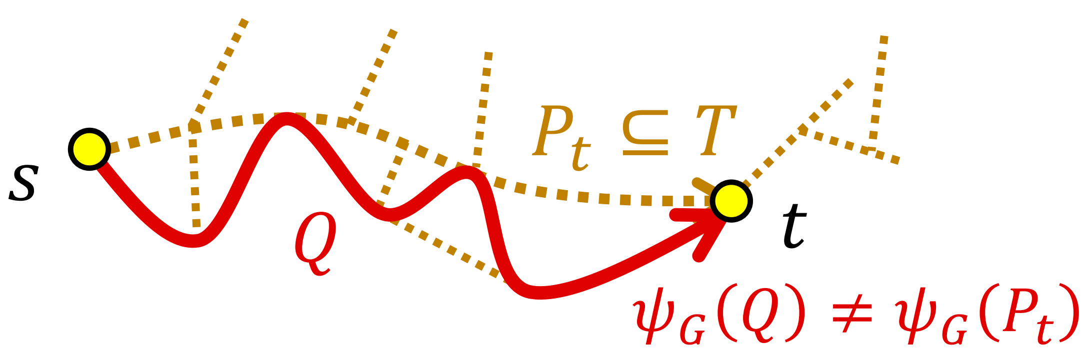
Goal Find a shortest **unorthodox** s–t path Q in G

1. Find a shortest s–t path P in G by **Dijkstra's Algorithm**

Output

Def.

A **Tree** in which each s–v path P_v is shortest in G



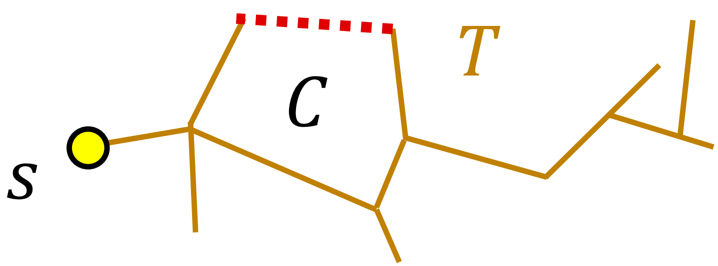
Finding a Shortest Unorthodox Path (SUP)

Input $(G = (V, E), \ell, s, t)$: Original Input

$T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s

Goal Find a shortest **unorthodox** s–t path Q in G

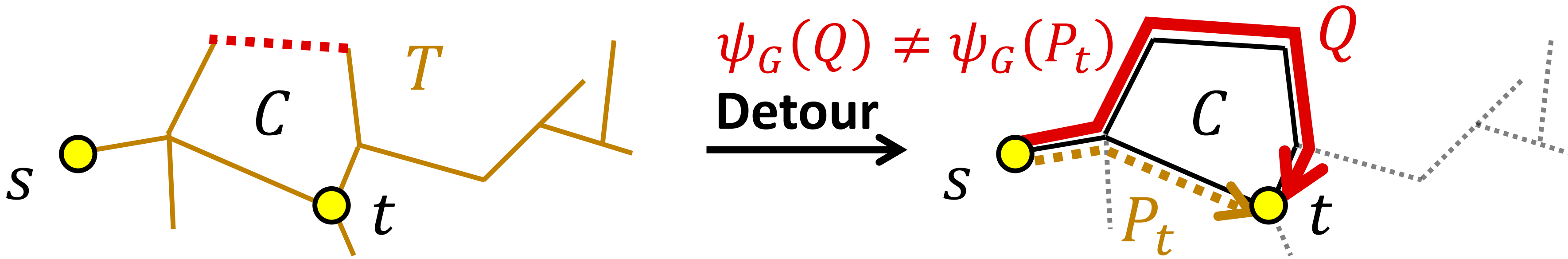
1. Find a "NICE" non-zero cycle C $(\psi_G(C) \neq 1_\Gamma)$
2. If t is on C, then return a **Detour** Q from P_t around C
3. Otherwise, shrink C into a single vertex b, and recursively solve a small instance



Finding a Shortest Unorthodox Path (SUP)

Input	$(G = (V, E), \ell, s, t)$: Original Input
	$T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s
Goal	Find a shortest **unorthodox** s–t path Q in G

1. Find a "NICE" non-zero cycle C $(\psi_G(C) \neq 1_\Gamma)$
2. If t is on C, then return a **Detour** Q from P_t around C
3. Otherwise, shrink C into a single vertex b, and recursively solve a small instance



Finding a Shortest Unorthodox Path (SUP)

Input $(G = (V, E), \ell, s, t)$: Original Input

$T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s

Goal Find a shortest **unorthodox** s–t path Q in G

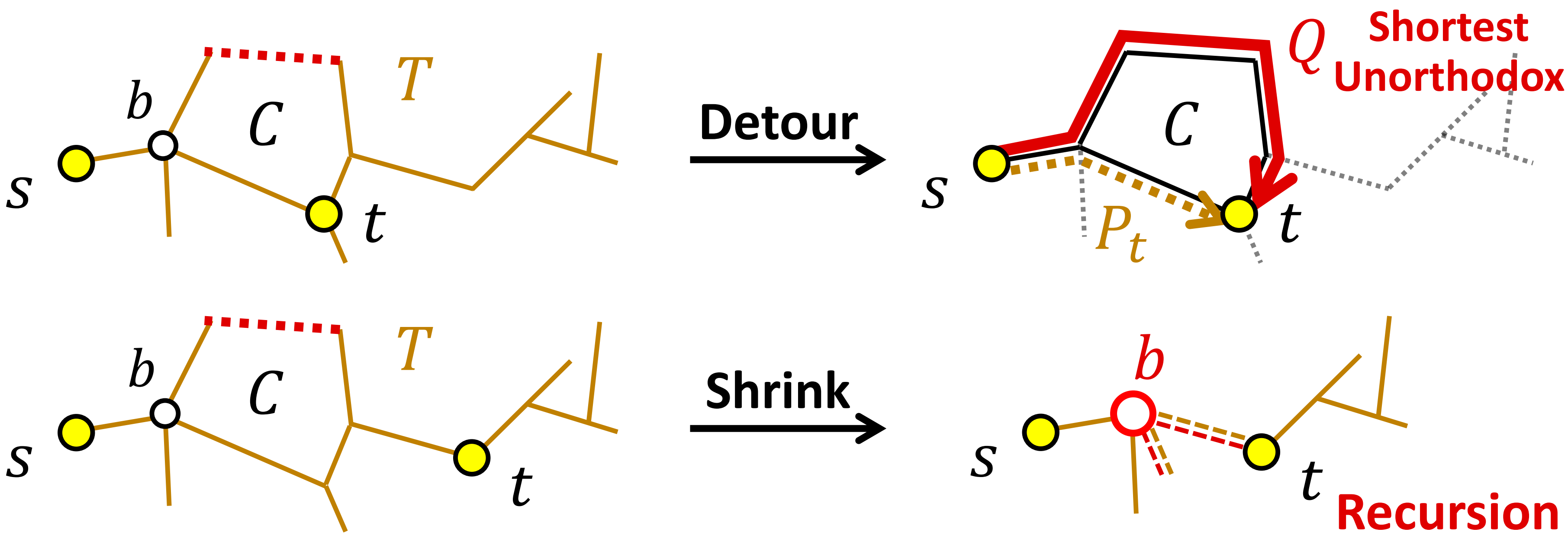
1. Find a "NICE" non-zero cycle C $(\psi_G(C) \neq 1_\Gamma)$
2. If t is on C, then return a **Detour** Q from P_t around C
3. Otherwise, shrink C into a single vertex b,
   and recursively solve a small instance

Main Lemma (Informal)

Lem. $\exists C$: Non-zero Cycle with a vertex $b \in C$ s.t.

- For a vertex in $C - b$, a detour Q around C is an **SUP**
- After shrinking C into b, **corresponding SUPs** remain

Moreover, such C can be found in $O(|E|)$ time



On Computational Time

Lem. $\exists C$: Non-zero Cycle with a vertex $b \in C$ s.t.

- For a vertex in $C - b$, a detour Q around C is an **SUP**
- After shrinking C into b, **corresponding SUPs** remain

Moreover, such C can be found in $\mathrm{O}(|E|)$ time

Obs. Shrinking occurs at most $(|V| - 2)$ times

Cor. An SUP can be found in $\mathrm{O}(|V| \cdot |E|)$ time (if exists)

Outline

- Algorithm Framework
  - Basic Idea
  - Auxiliary Problem (Shortest Unorthodox Path)
  - Main Lemma
- Key Structure: Lowest Blossoms
  - Detour yields a Shortest Unorthodox Path (SUP)
  - Shrinking preserves SUP Problem
- Conclusion

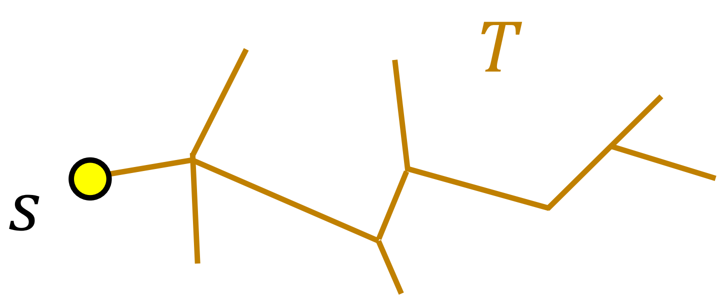
Blossom

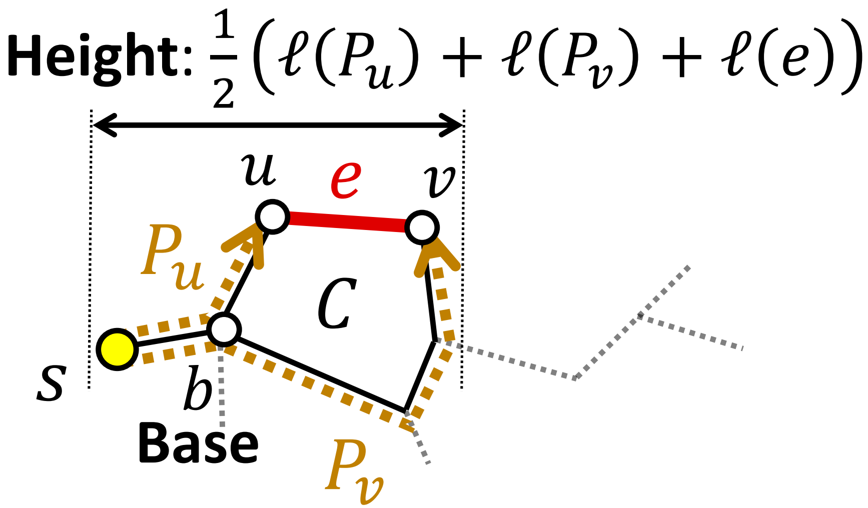
Def. $T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s

C is a **Blossom**

$$\Updownarrow$$

- $\exists e \in E - T$ s.t. $C \subseteq T + e$ (i.e., C is a Fundamental Circuit)
- $\psi_G(C) \neq 1_\Gamma$ $\left(\Leftrightarrow \psi_G(P_u * e) \neq \psi_G(P_v)\right)$





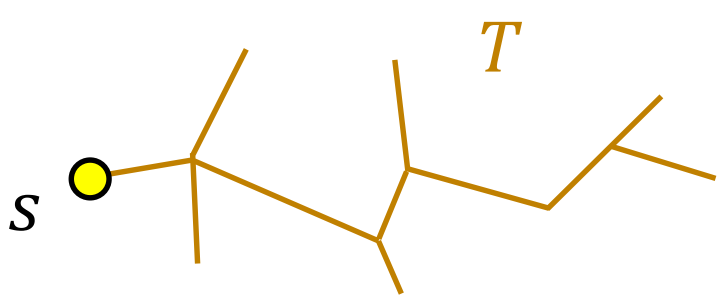
Lowest Blossom (LB)

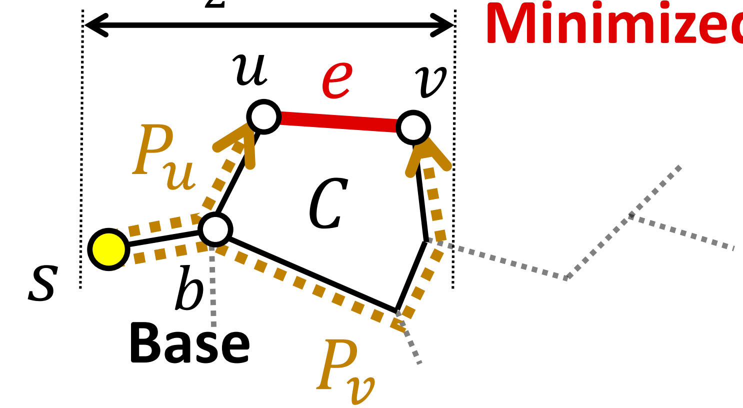
Def. $T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s

C is a **Lowest Blossom**

$$\Updownarrow$$

- $\exists e \in E - T$ s.t. $C \subseteq T + e$ (i.e., C is a Fundamental Circuit)
- $e \in \underset{f = \{x, y\}}{\arg\min} \{\ell(P_x) + \ell(P_y) + \ell(f) \mid \psi_G(P_x * f) \neq \psi_G(P_y)\}$





Detour around LB yields SUP

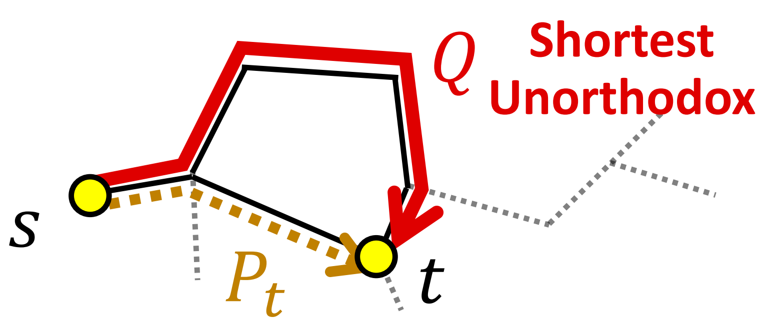
Def. $T = \bigcup_{v \in V} P_v$: Shortest Path Tree of G rooted at s

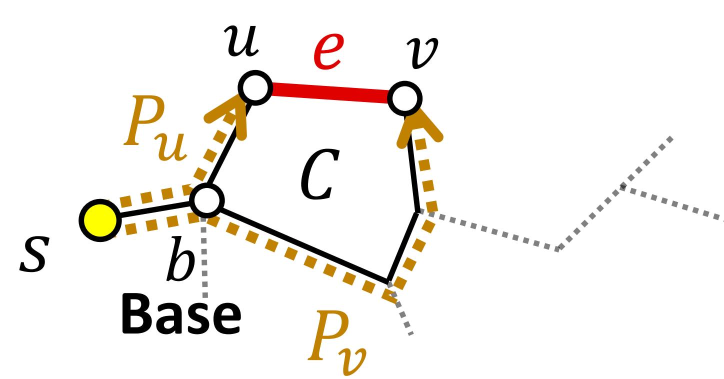
C is a **Lowest Blossom**

$$\Updownarrow$$

- $\exists e \in E - T$ s.t. $C \subseteq T + e$ (i.e., C is a Fundamental Circuit)
- $e \in \arg\min_{f = \{x,y\}} \{\ell(P_x) + \ell(P_y) + \ell(f) \mid \psi_G(P_x * f) \neq \psi_G(P_y)\}$

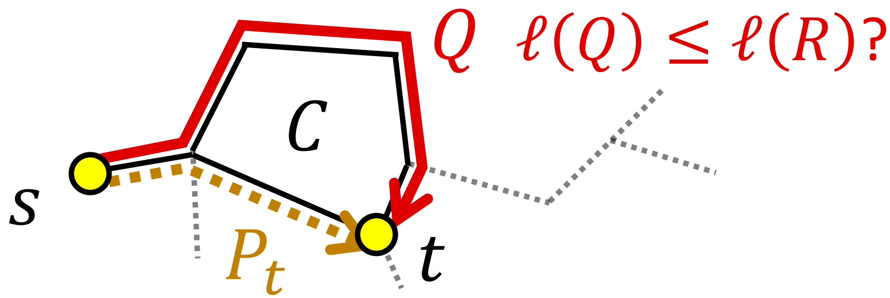
Lem. For $C - b$, every detour Q from T is an **SUP**

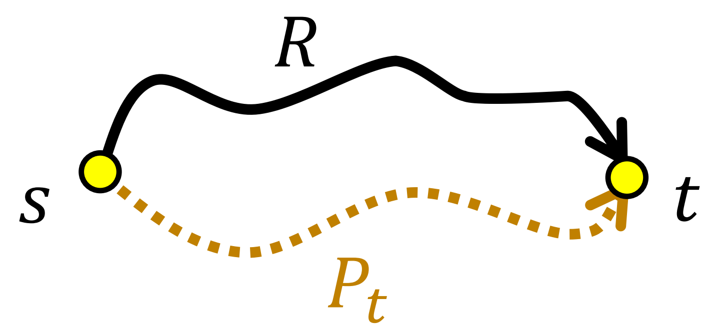




Detour around LB yields SUP

Lem. For $C - b$, every detour Q from T is an **SUP**



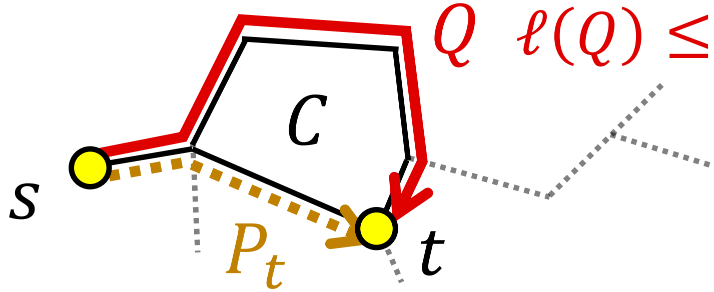


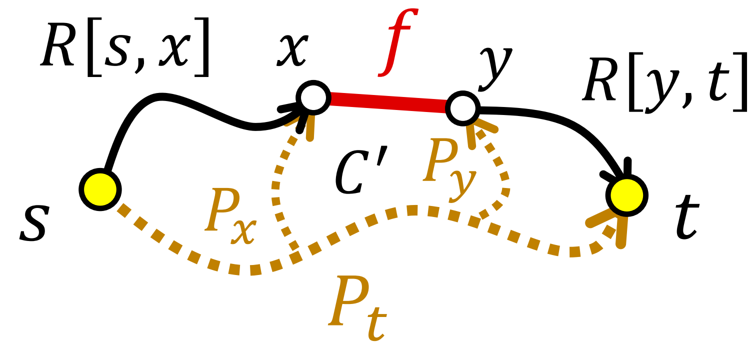
Fix an s–t path R with $\psi_G(R) \neq \psi_G(P_t)$

- $\exists f = \{x, y\} \in R$ s.t. $\psi_G(P_x * f) \neq \psi(P_y)$
  $\rightarrow \ell(P_x) + \ell(P_y) + \ell(f) \geq \ell(P_t) + \ell(Q)$ (C is an LB)
- $\ell(R[s, x]) \geq \ell(P_x)$ (P_x is shortest)
- $\ell(R[y, t]) \geq |\ell(P_y) - \ell(P_t)|$ (o/w, $\exists$shortcut for T)

Detour around LB yields SUP

Lem. For $C - b$, every detour Q from T is an **SUP**





Fix an s–t path R with $\psi_G(R) \neq \psi_G(P_t)$

- $\exists f = \{x,y\} \in R$ s.t. $\psi_G(P_x * f) \neq \psi(P_y)$

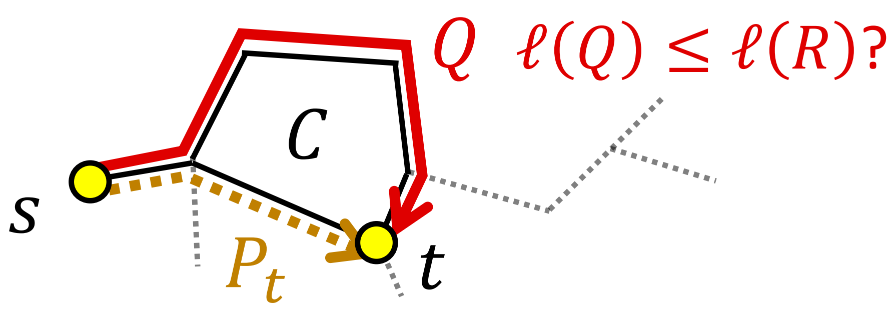
o/w, $\psi_G(R) = \prod_{f=\{x,y\}\in R} \left(\psi_G(P_x)^{-1} \cdot \psi_G(P_y)\right)$

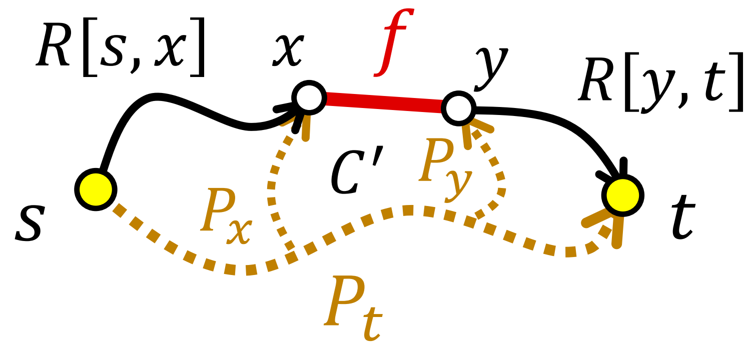
$$= \psi_G(P_s)^{-1} \cdot \psi_G(P_t) = \psi_G(P_t)$$

Contradiction!

Detour around LB yields SUP

Lem. For $C - b$, every detour Q from T is an **SUP**



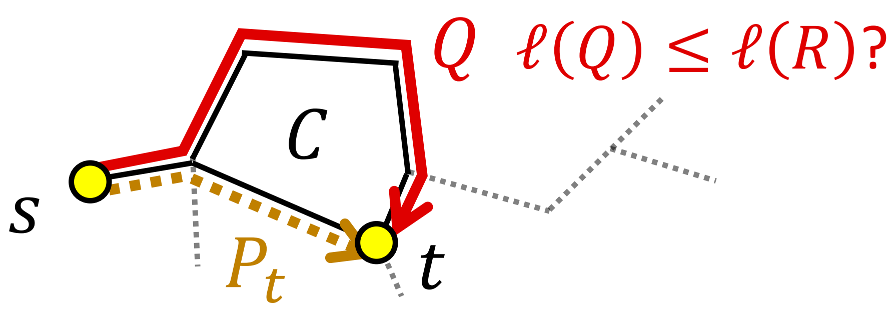


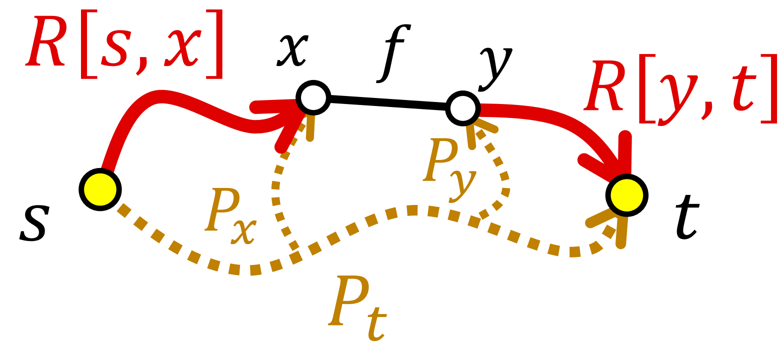
Fix an s–t path R with $\psi_G(R) \neq \psi_G(P_t)$

- $\exists f = \{x, y\} \in R$ s.t. $\psi_G(P_x * f) \neq \psi(P_y)$
  $\rightarrow \ell(P_x) + \ell(P_y) + \ell(f) \geq \ell(P_t) + \ell(Q)$ (C is an LB)

Detour around LB yields SUP

Lem. For $C - b$, every detour Q from T is an **SUP**



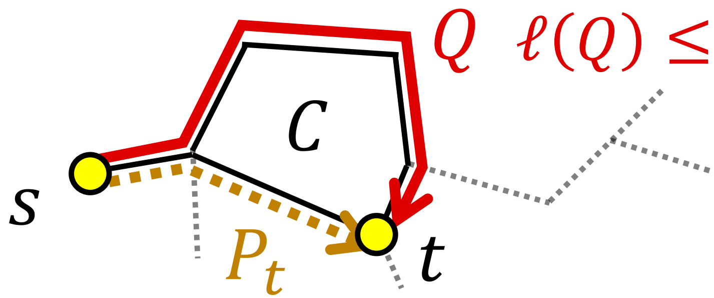


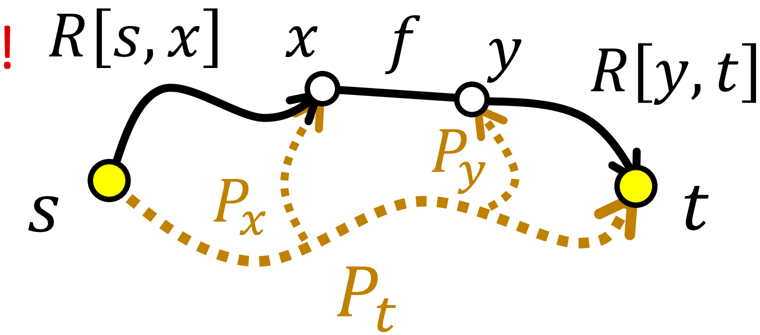
Fix an s–t path R with $\psi_G(R) \neq \psi_G(P_t)$

- $\exists f = \{x, y\} \in R$ s.t. $\psi_G(P_x * f) \neq \psi(P_y)$
  $\rightarrow \ell(P_x) + \ell(P_y) + \ell(f) \geq \ell(P_t) + \ell(Q)$ (C is an LB)
- $\ell(R[s,x]) \geq \ell(P_x)$ (P_x is shortest)
- $\ell(R[y,t]) \geq \left|\ell(P_y) - \ell(P_t)\right|$ (o/w, $\exists$shortcut for T)

Detour around LB yields SUP

<u>Lem.</u> For $C - b$, every detour Q from T is an **SUP**





$$\boxed{\ell(R) \geq \ell(P_x) + \ell(f) + \ell(P_y) - \ell(P_t) \geq \ell(Q)}$$

$\ell(P_x) + \ell(P_y) + \ell(f) \geq \ell(P_t) + \ell(Q)$ (C is an LB)

- $\ell(R[s,x]) \geq \ell(P_x)$ (P_x is shortest)
- $\ell(R[y,t]) \geq \left|\ell(P_y) - \ell(P_t)\right|$ (o/w, ∃shortcut for T)

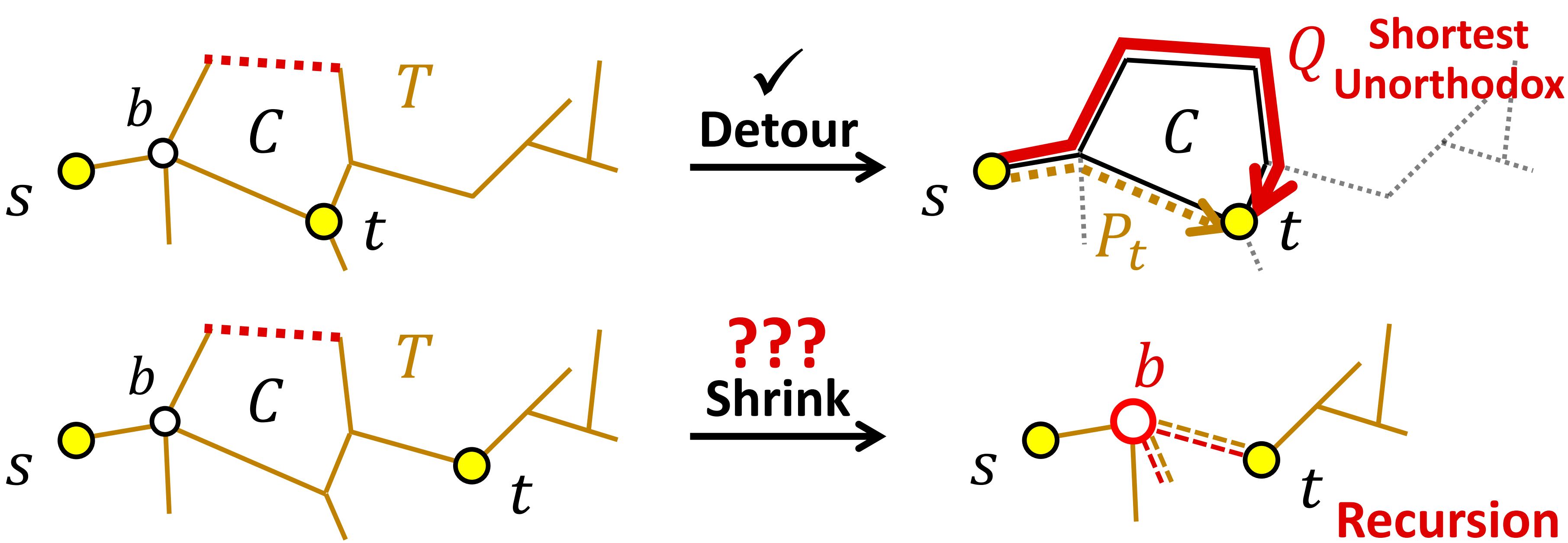
Main Lemma (Informal)

Lem. $\exists C$: Non-zero Cycle with a vertex $b \in C$ s.t.

✓ For a vertex in $C - b$, a detour Q around C is an **SUP**

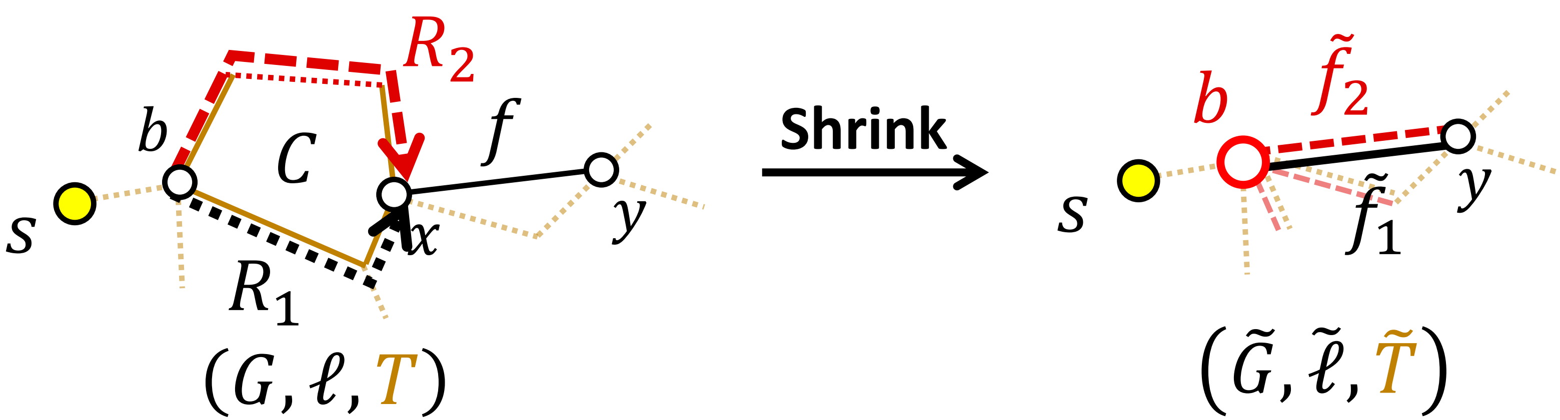
? After shrinking C into b, **corresponding SUPs** remain

✓ Moreover, such C can be found in $O(|E|)$ time



Shrinking preserves SUP Problem

Lem. $\forall R$: Unorthodox s–t path in G intersecting C,
$\exists R'$: Unorthodox s–t path in G s.t. $\ell(R') \leq \ell(R)$
and R' remains in a shrunk form



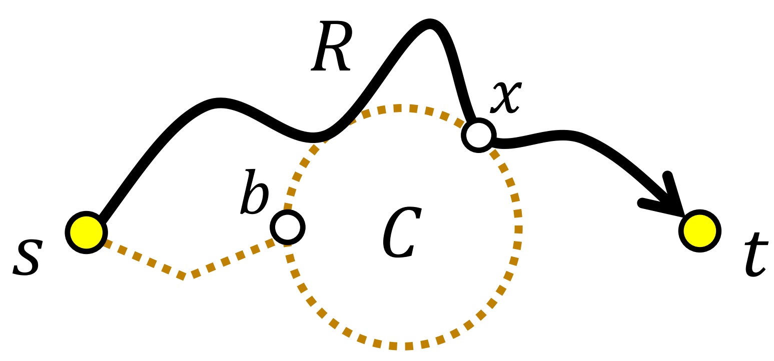
$$\psi_{\tilde{G}}(\tilde{f}_i; b \to y) := \psi_G(R_i * f)$$
$$\tilde{\ell}(\tilde{f}_i) := \ell(R_i) + \ell(f)$$

Shrinking preserves SUP Problem

Lem. $\forall R$: Unorthodox s–t path in G intersecting C,
$\exists R'$: Unorthodox s–t path in G s.t. $\ell(R') \leq \ell(R)$
and R' remains in a shrunk form

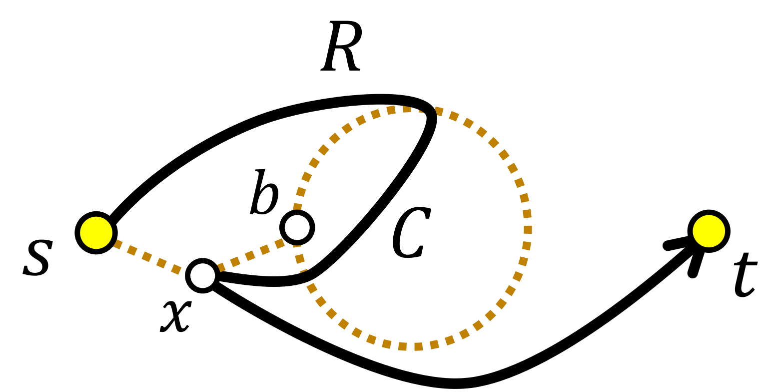
R: Unorthodox s–t path, x: Last Vertex intersecting $P_b \cup C$

Case 1. $x \in C - b$ (Easy)



$\psi_G(R) \neq \psi_G(P_t)$

Case 2. $x \in P_b$ (Complicated)

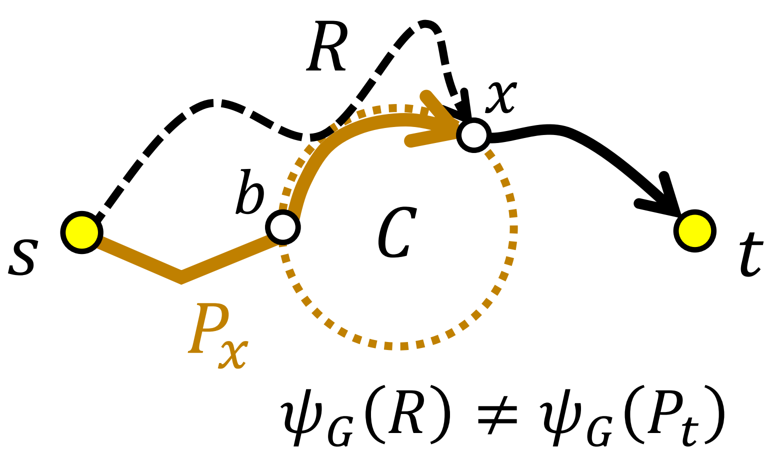


Shrinking preserves SUP Problem

Lem. $\forall R$: Unorthodox s–t path in G intersecting C,
$\exists R'$: Unorthodox s–t path in G s.t. $\ell(R') \leq \ell(R)$
and R' remains in a shrunk form

R: Unorthodox s–t path, x: Last Vertex intersecting $P_b \cup C$

Case 1. $x \in C - b$ (Easy)



$$\psi_G(P_x * R[x,t]) \neq \psi_G(P_t)$$
$$\Downarrow$$
$R' \coloneqq P_x * R[x,t]$ is OK

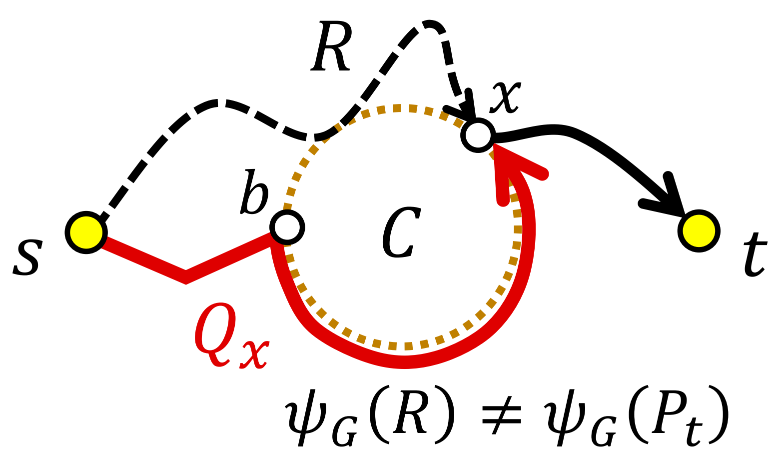
- $\ell(P_x) \leq \ell(R[s,x])$

Shrinking preserves SUP Problem

Lem. $\forall R$: Unorthodox s–t path in G intersecting C,
$\exists R'$: Unorthodox s–t path in G s.t. $\ell(R') \leq \ell(R)$
and R' remains in a shrunk form

R: Unorthodox s–t path, x: Last Vertex intersecting $P_b \cup C$

Case 1. $x \in C - b$ (Easy)



$$\psi_G(P_x * R[x,t]) = \psi_G(P_t)$$
$$\Downarrow$$
$R' := Q_x * R[x,t]$ is OK

- $\psi_G(Q_x) \neq \psi_G(P_x) \neq \psi_G(R[s,x])$
- $\ell(Q_x) \leq \ell(R[s,x])$

Outline

- Algorithm Framework
  - Basic Idea
  - Auxiliary Problem (Shortest Unorthodox Path)
  - Main Lemma
- Key Structure: Lowest Blossoms
  - Detour yields a Shortest Unorthodox Path (SUP)
  - Shrinking preserves SUP Problem
- Conclusion

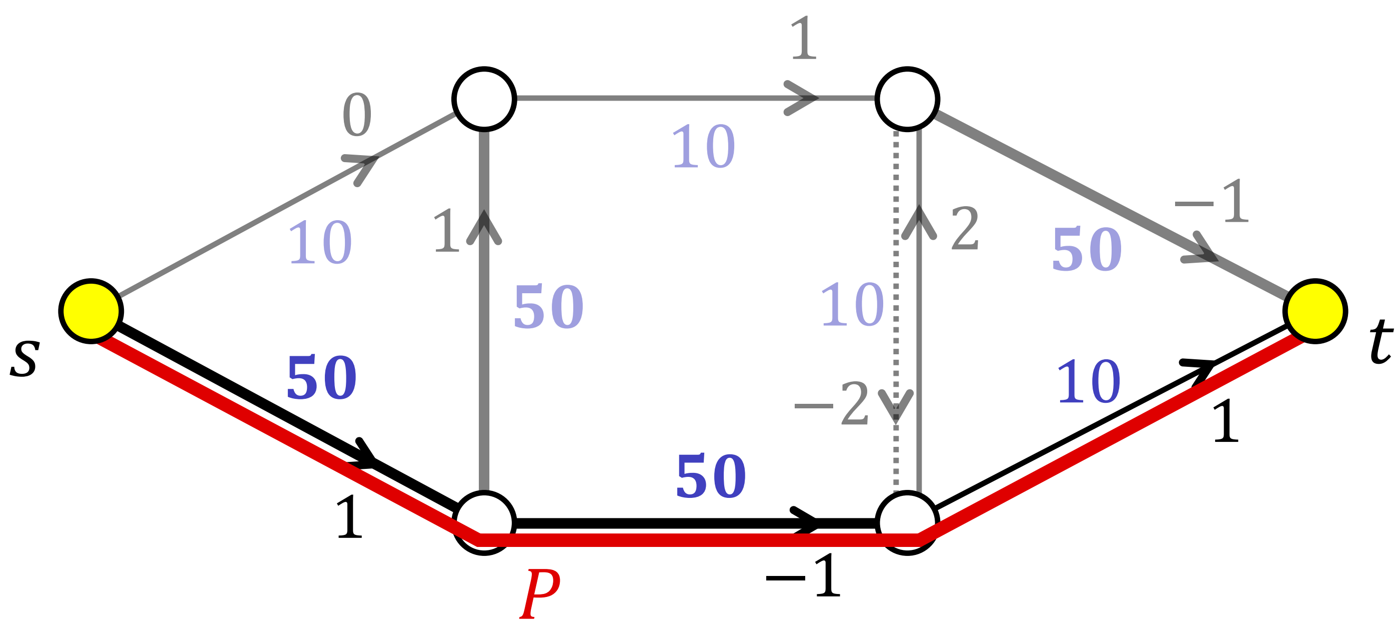
Conclusion

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}^E_{\geq 0}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G

Thm. Solved by $O(|V| \cdot |E|)$ Elementary Operations



Conclusion

Input $G = (V, E)$: Γ-Labeled Graph (Γ: Group)

$\ell \in \mathbf{R}_{\geq 0}^{E}$: Edge Length, $s, t \in V$: Terminals

Goal Find a shortest **non-zero** s–t path P in G

Thm. Solved by $O(|V| \cdot |E|)$ Elementary Operations

- Dijkstra + Shrinking **Lowest Blossoms**
- Depending heavily on **Nonnegativity of Edge Length**

Q. How about a general input "without **Negative Cycle**"?

[Unconstrained] **Strongly-Poly** via **Weighted Matching**

[Parity Constrained] Open