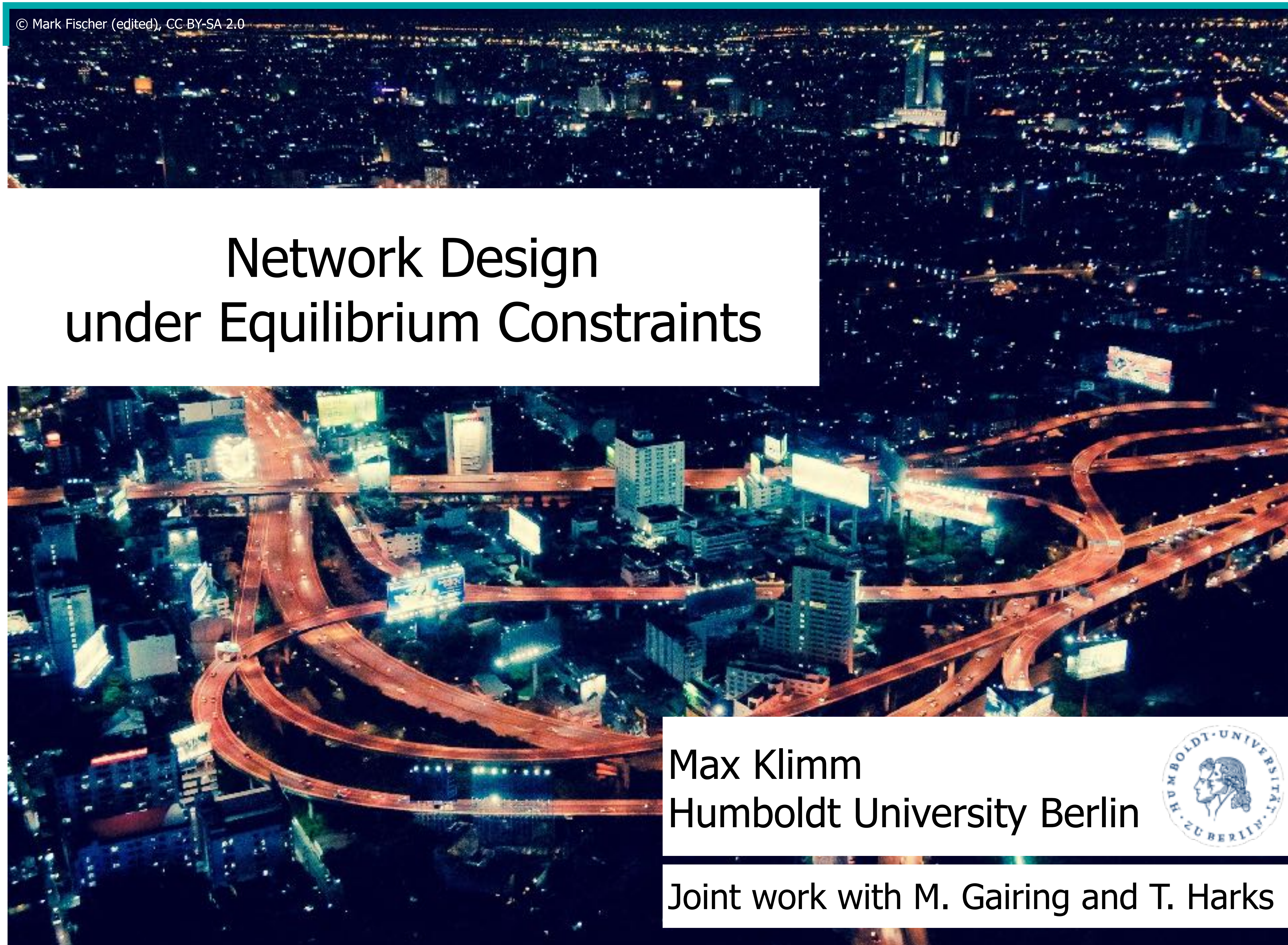


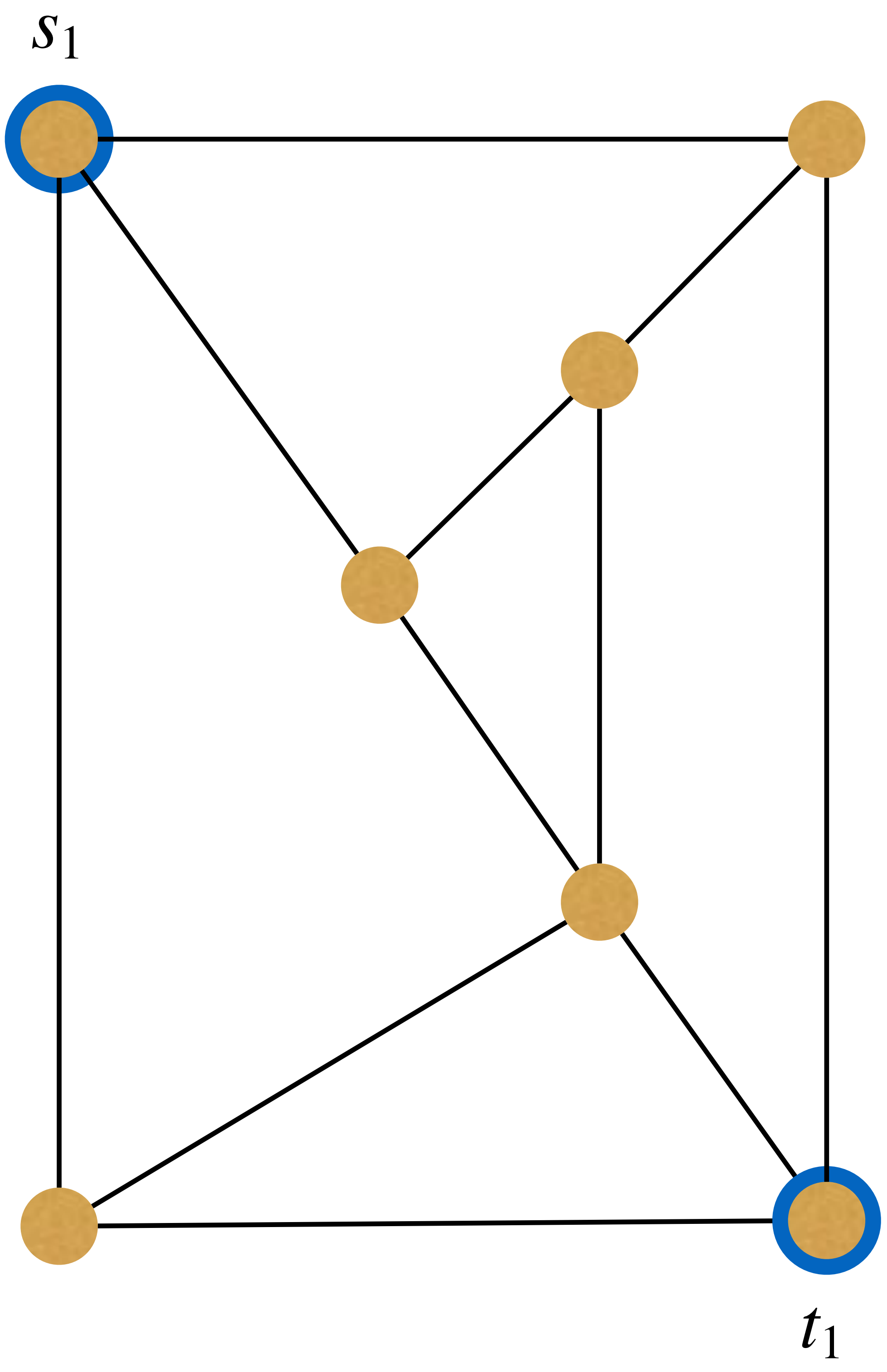
Network Design under Equilibrium Constraints

Max Klimm
Humboldt University Berlin

Joint work with M. Gairing and T. Harks

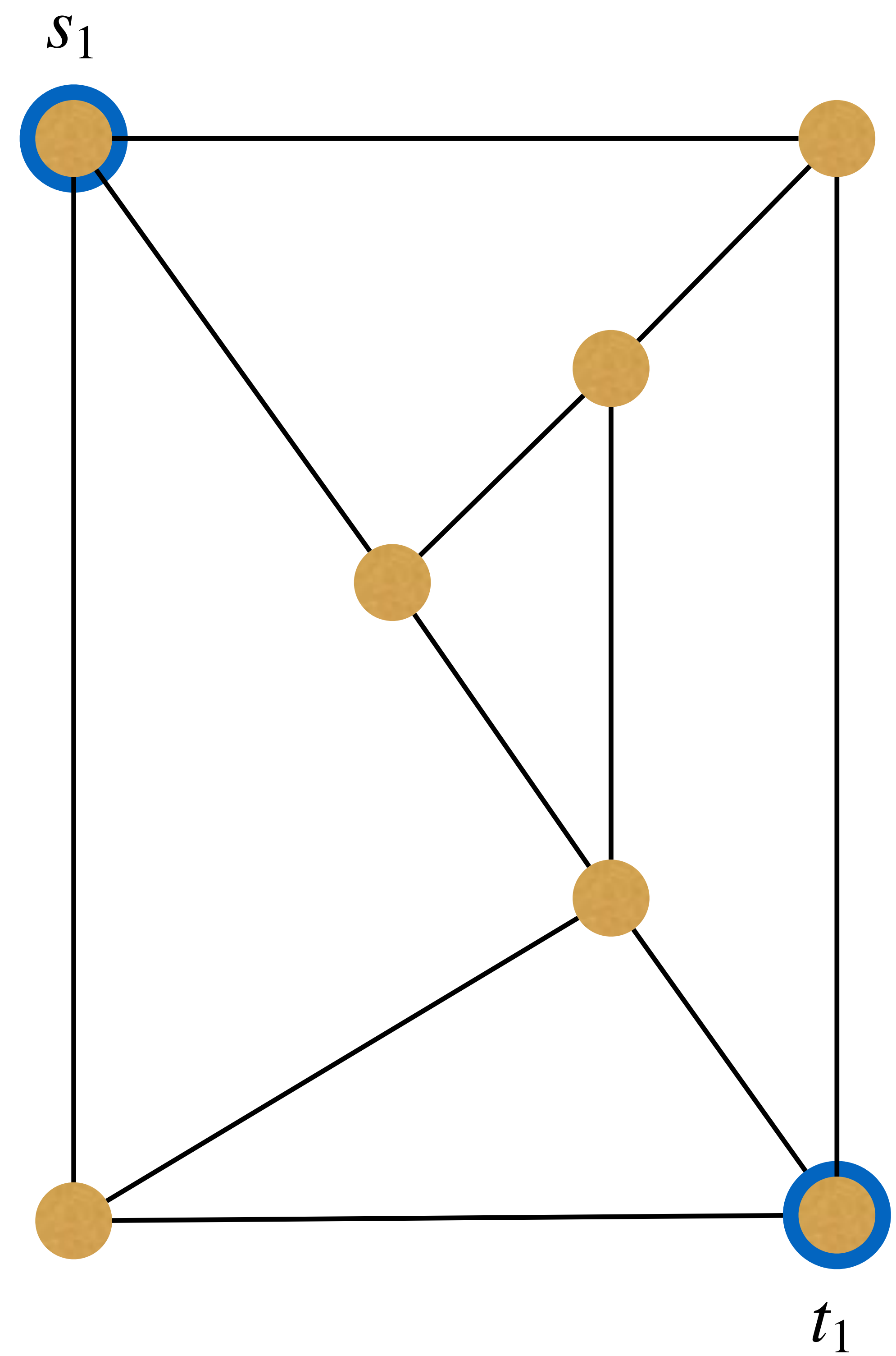
Selfish flows

- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \to \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in \mathrm{V} \times \mathrm{V} \times \mathbb{R}_+$



Selfish flows

- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \to \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in V \times V \times \mathbb{R}_+$



Definition — Wardrop equilibrium
Multicommodity flow $\mathbf{f} = (f_{i,P})$ with

$$\sum_{e \in P} c_e(f_e) \leq \sum_{e \in Q} c_e(f_e)$$

for all (s_i,t_i)-paths P,Q with $f_{i,P} > 0$.

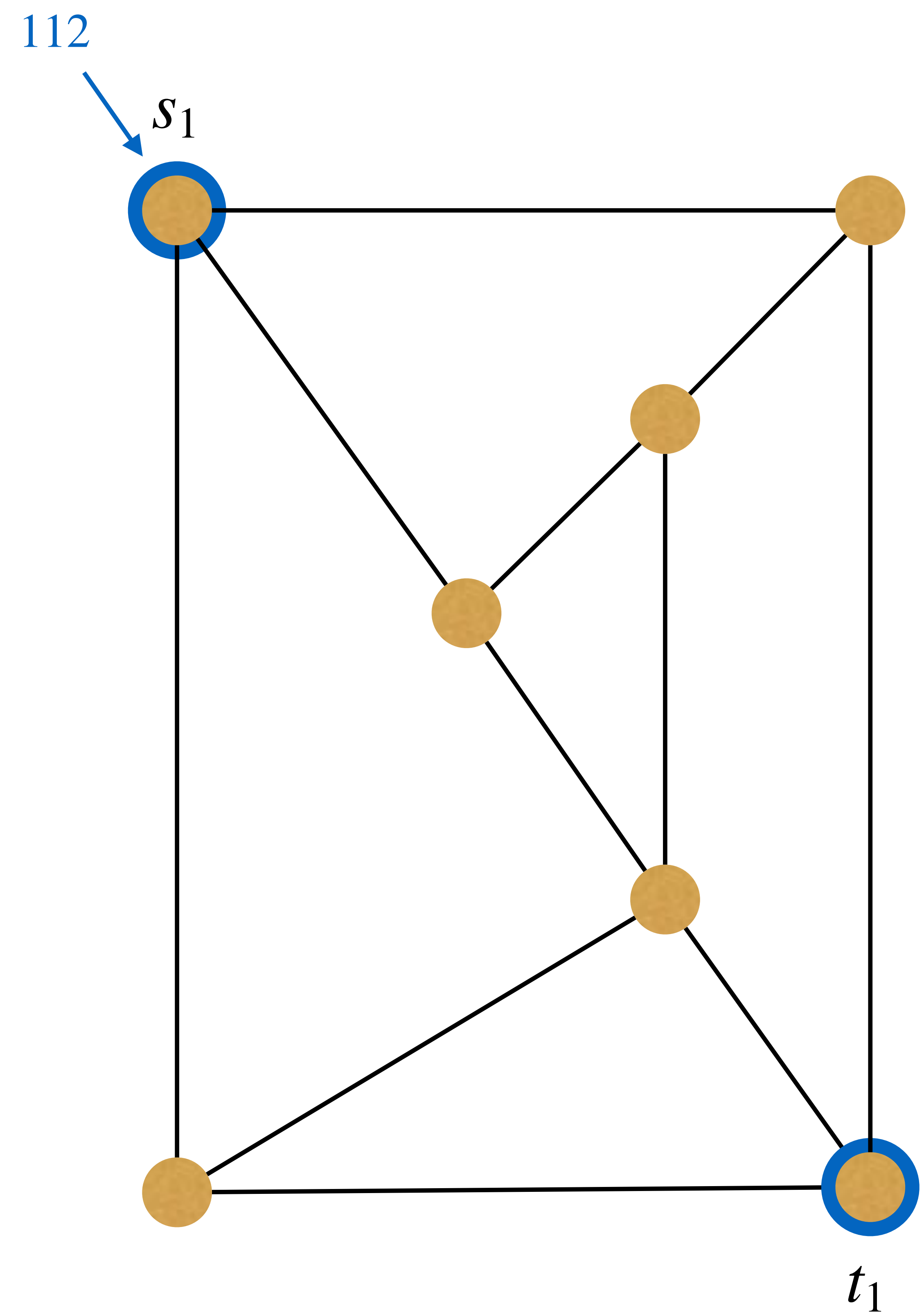
Selfish flows

- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \to \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in V \times V \times \mathbb{R}_+$

Definition — Wardrop equilibrium
Multicommodity flow $\mathbf{f} = (f_{i,P})$ with

$$\sum_{e \in P} c_e(f_e) \leq \sum_{e \in Q} c_e(f_e)$$

for all (s_i,t_i)-paths P,Q with $f_{i,P} > 0$.



$c_e(x) = x$ for all e

Selfish flows

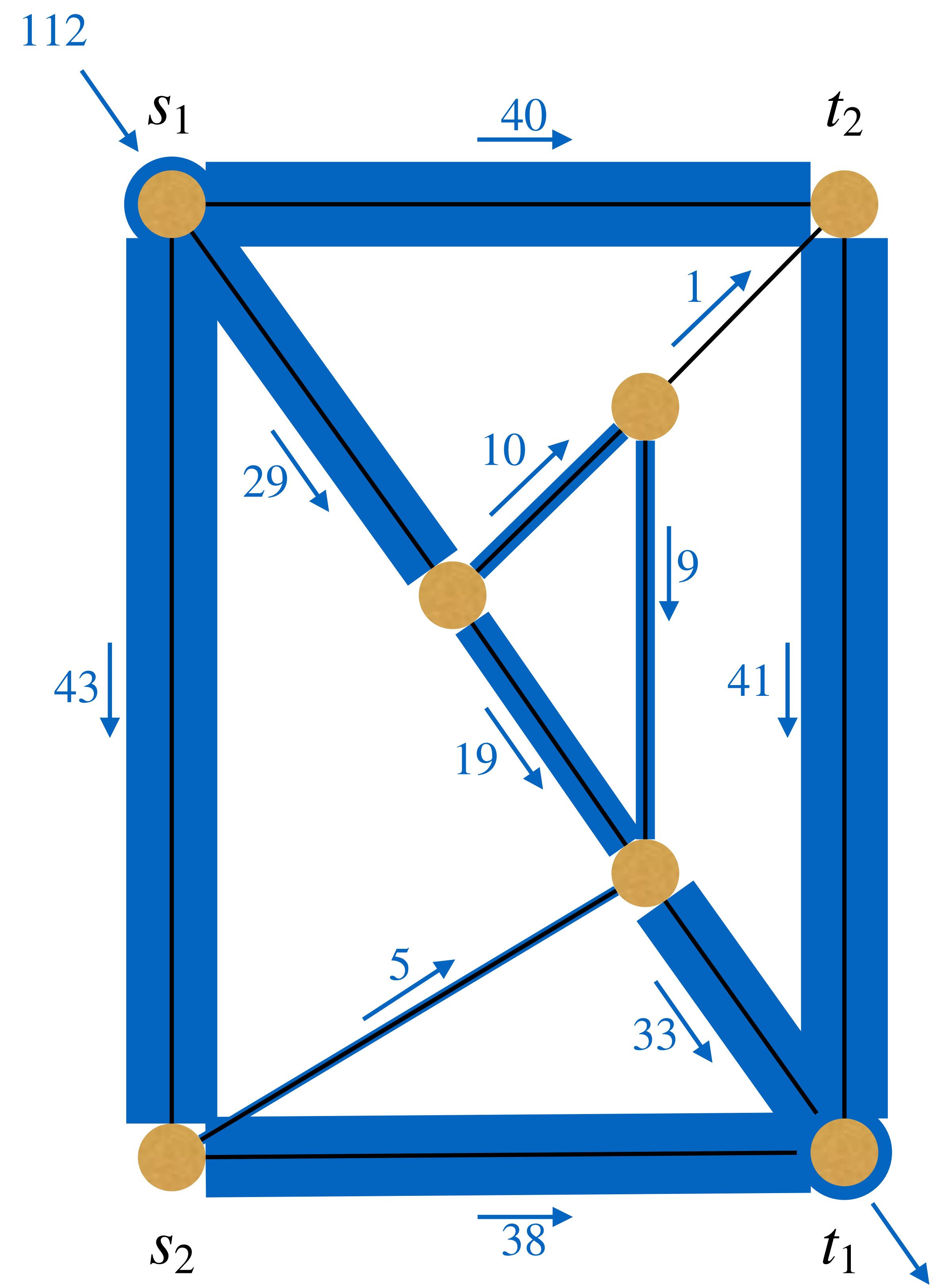
- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \to \mathbb{R}_+$ (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in V \times V \times \mathbb{R}_+$

Definition — Wardrop equilibrium

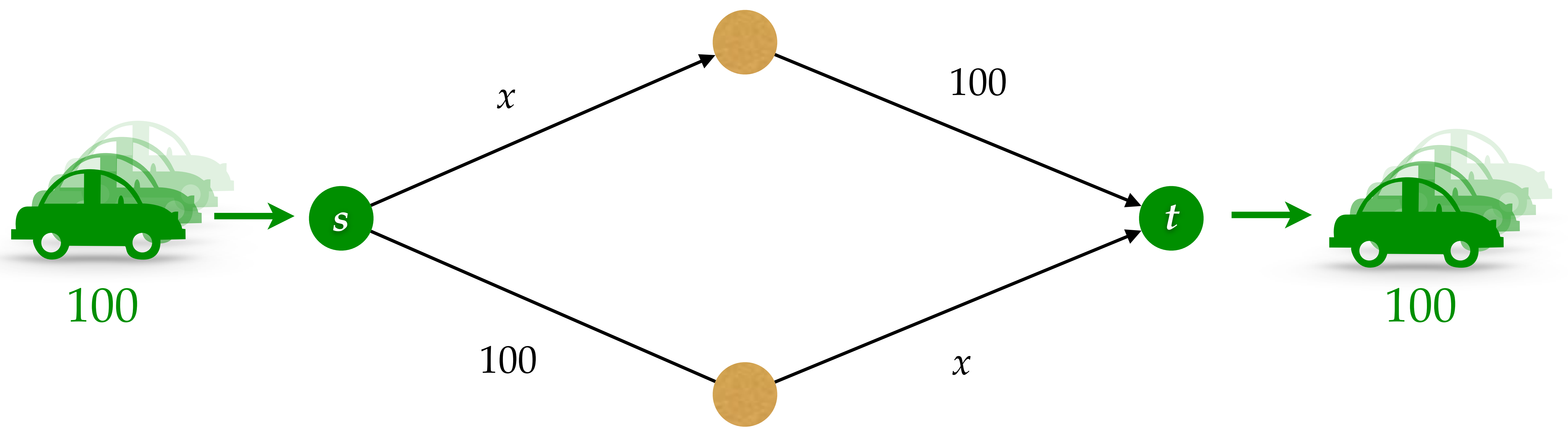
Multicommodity flow $\mathbf{f} = (f_{i,P})$ with

$$\sum_{e \in P} c_e(f_e) \leq \sum_{e \in Q} c_e(f_e)$$

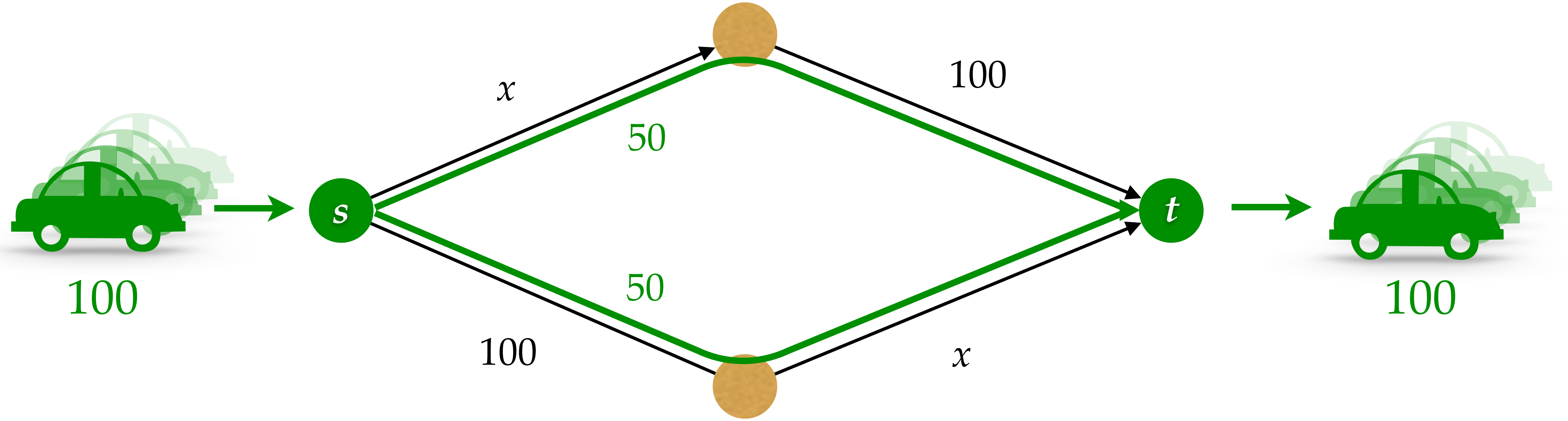
for all (s_i,t_i)-paths P,Q with $f_{i,P} > 0$.



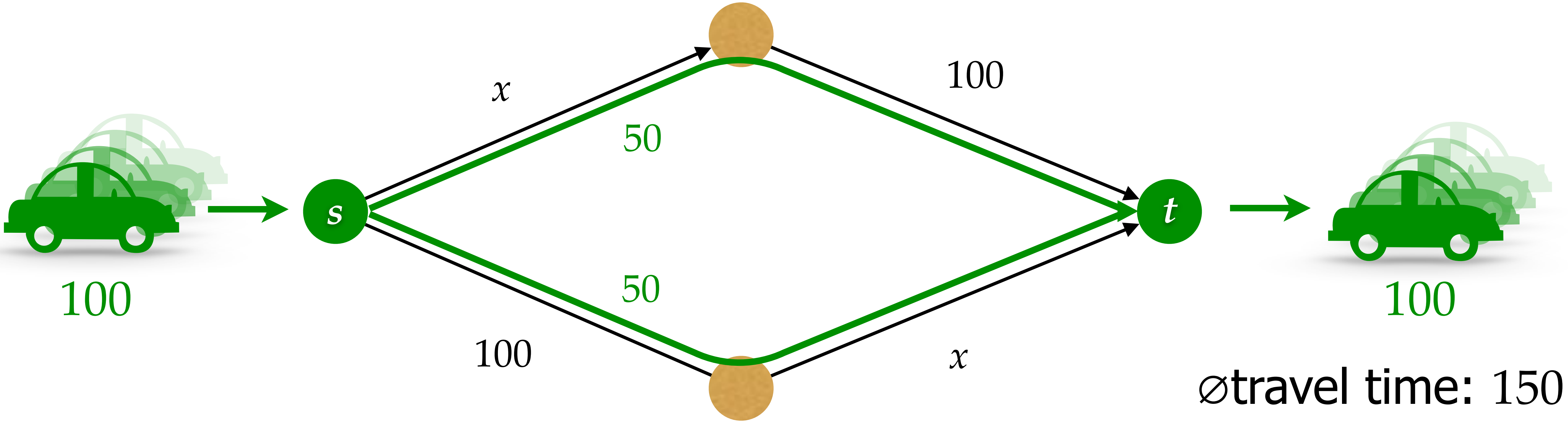
Braess' Paradox



Braess' Paradox

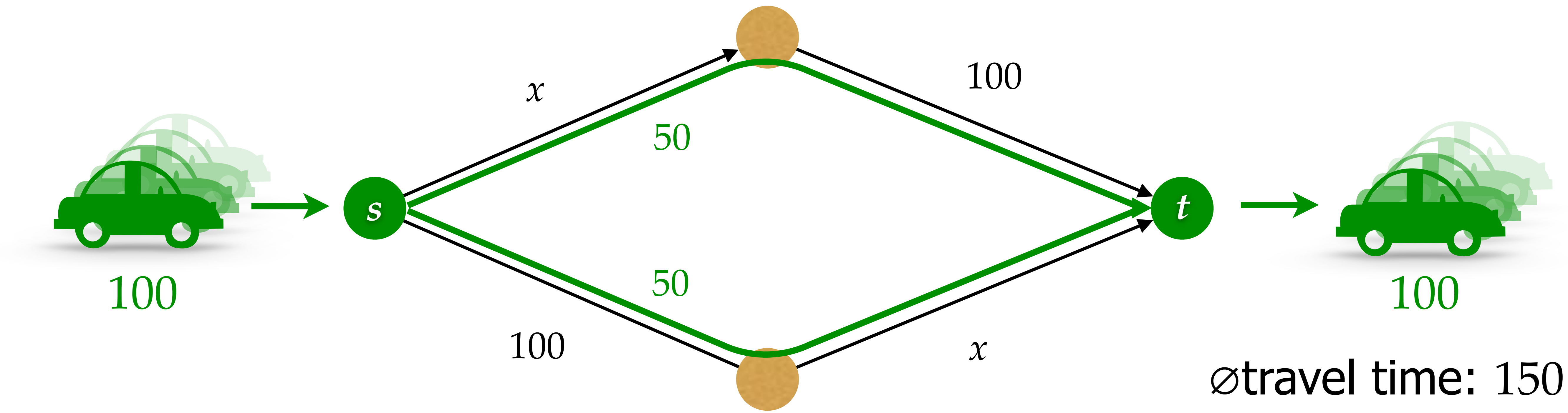


Braess' Paradox

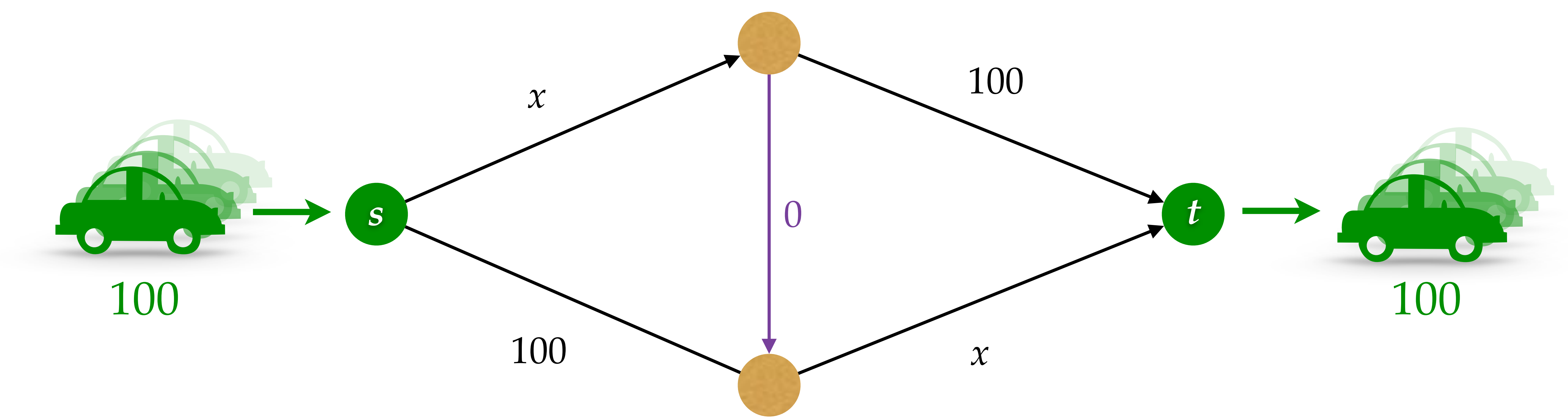


∅travel time: 150

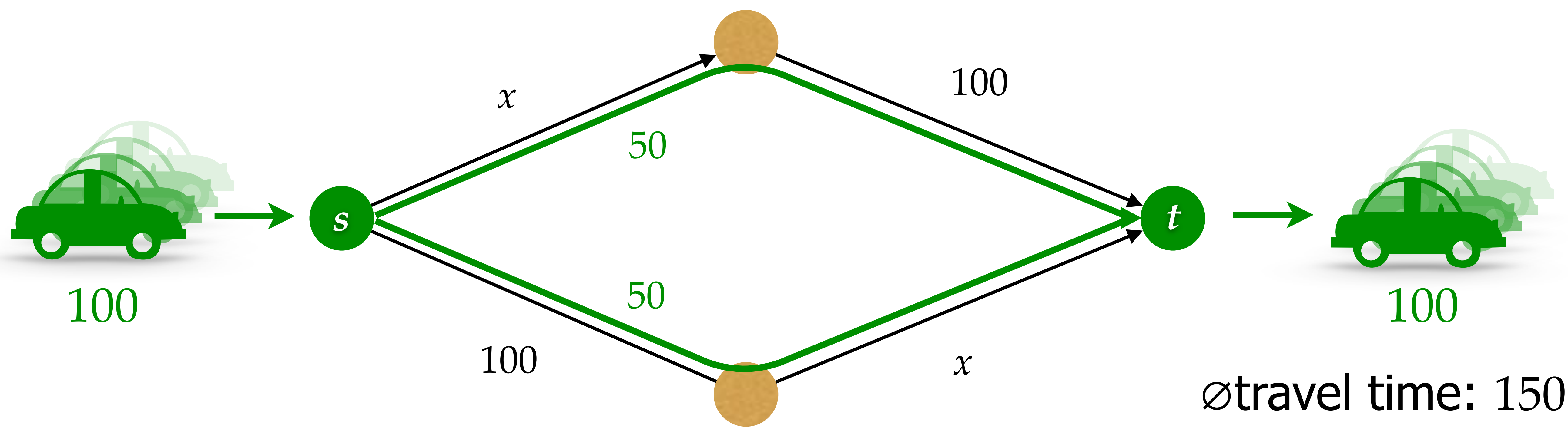
Braess' Paradox



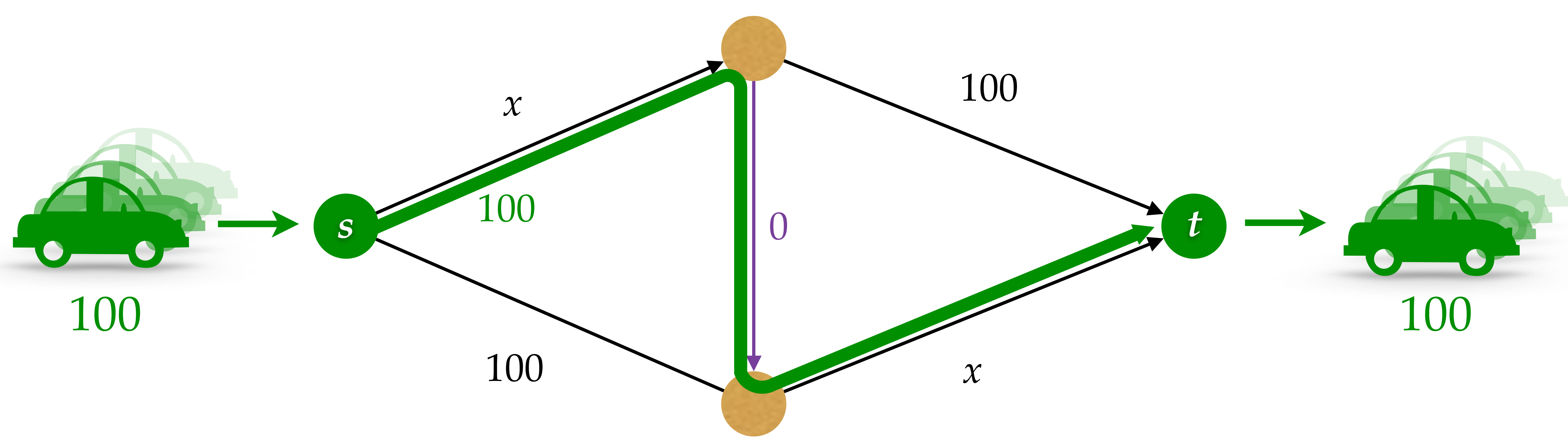
⌀travel time: 150



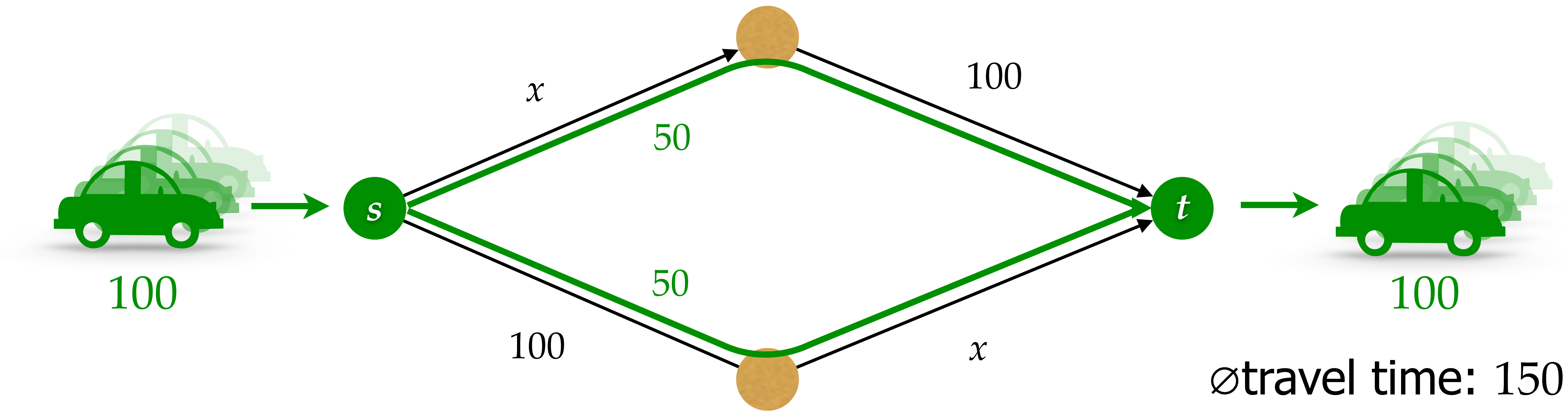
Braess' Paradox



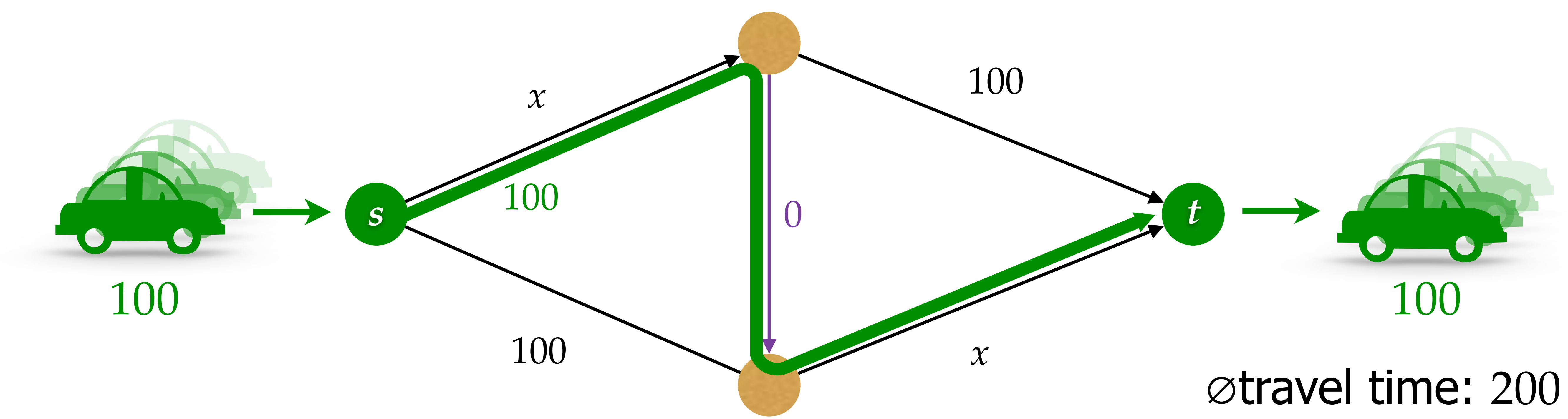
⌀travel time: 150



Braess' Paradox



⌀travel time: 150

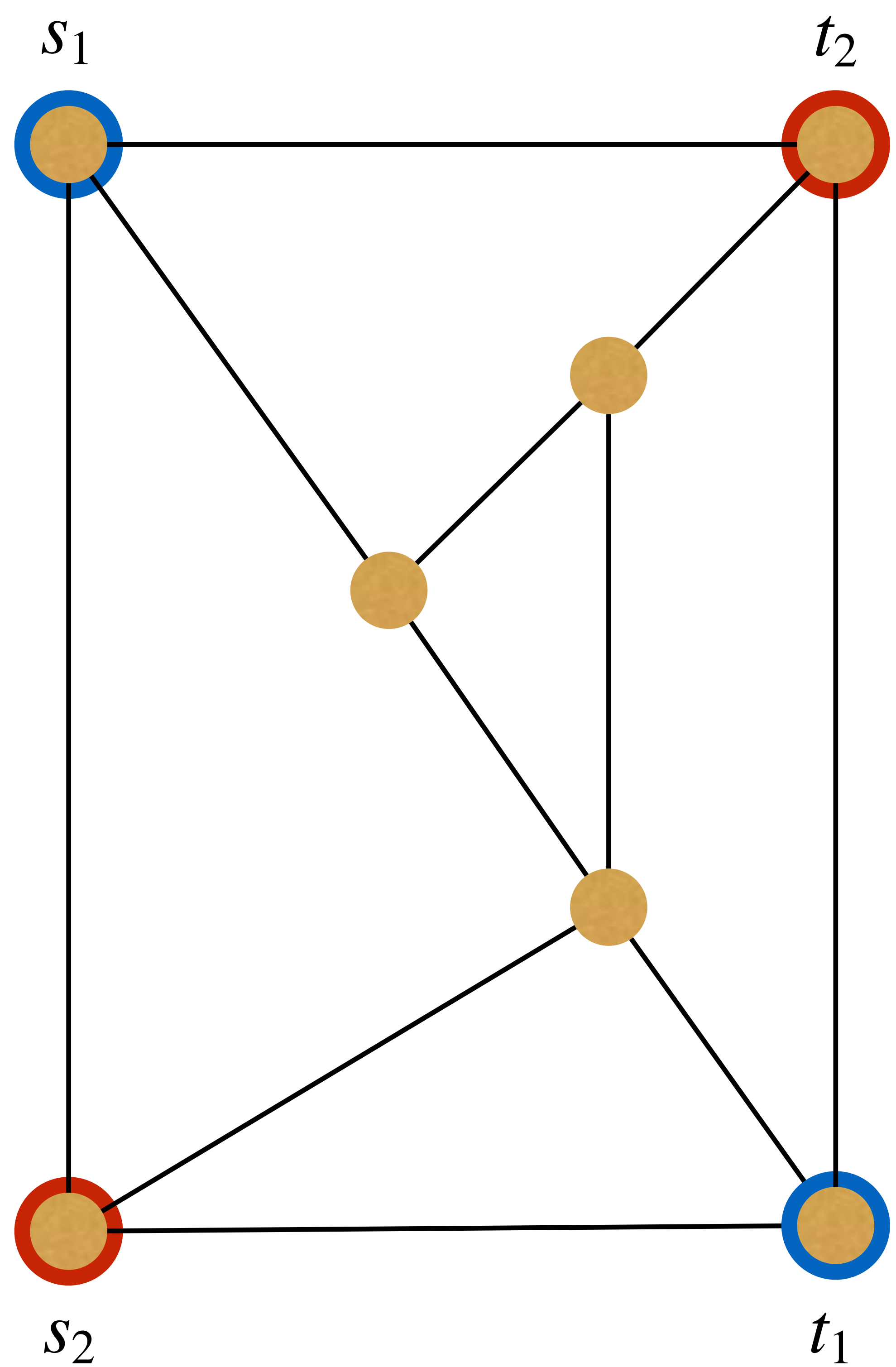


⌀travel time: 200

How can road networks be designed so that good equilibria emerge?

Continuous network design problem

- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \rightarrow \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in \mathrm{V} \times \mathrm{V} \times \mathbb{R}_+$
  - construction costs $k_e \in \mathbb{R}_+$



Continuous network design problem

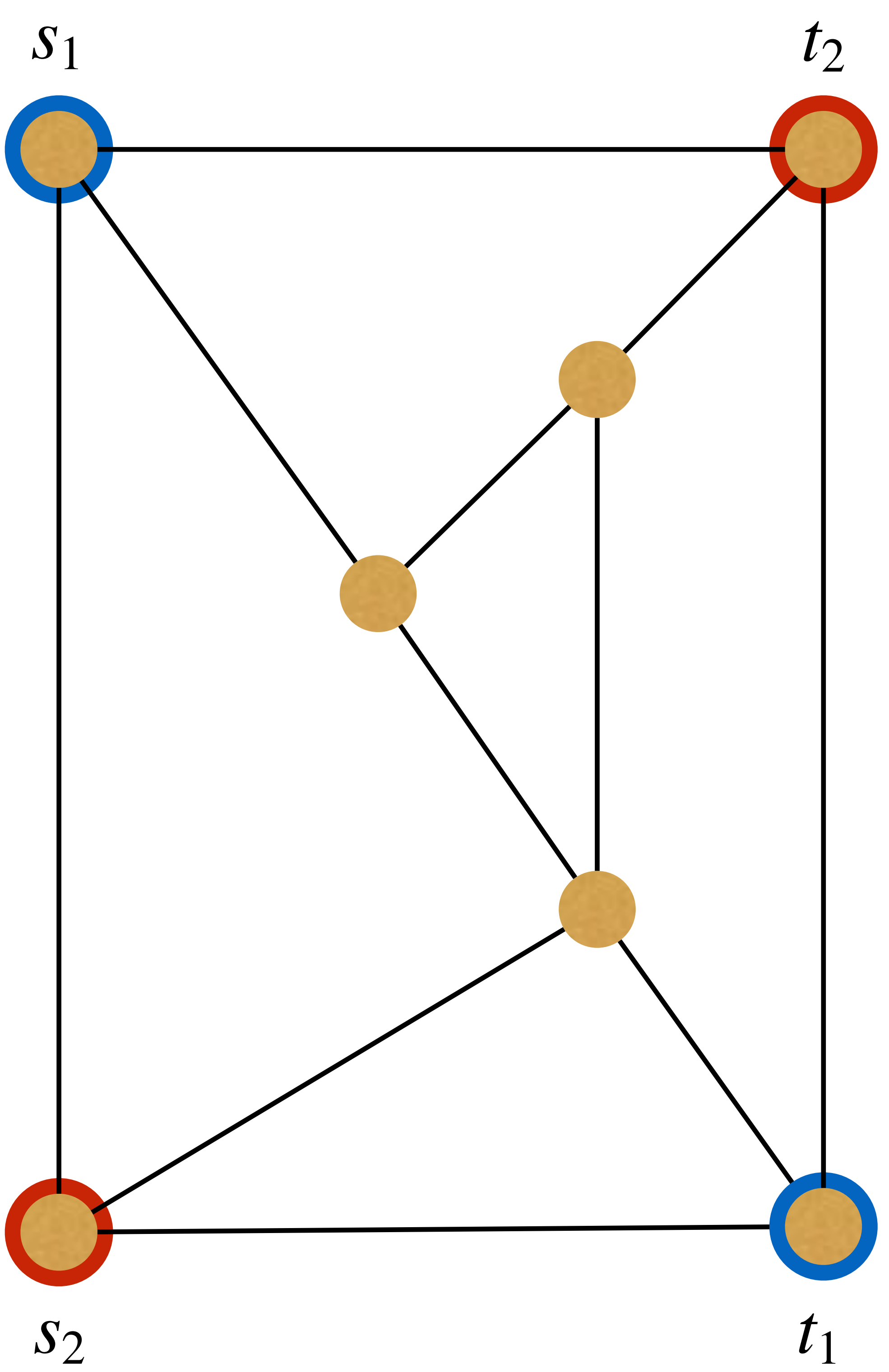
- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \rightarrow \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in \mathrm{V} \times \mathrm{V} \times \mathbb{R}_+$
  - construction costs $k_e \in \mathbb{R}_+$

Continuous network design problem

$$\min_{\mathbf{f},\, \mathbf{z}} \sum_{e \in E} \left(c_e(f_e / z_e) f_e + z_e k_e \right)$$

s.t.: $\mathbf{z} = (z_e)_{e \in E}$, $z_e \in \mathbb{R}_+$

$\mathbf{f} = (f_e)_{e \in E}$ is a Wardrop equilibrium



Continuous network design problem

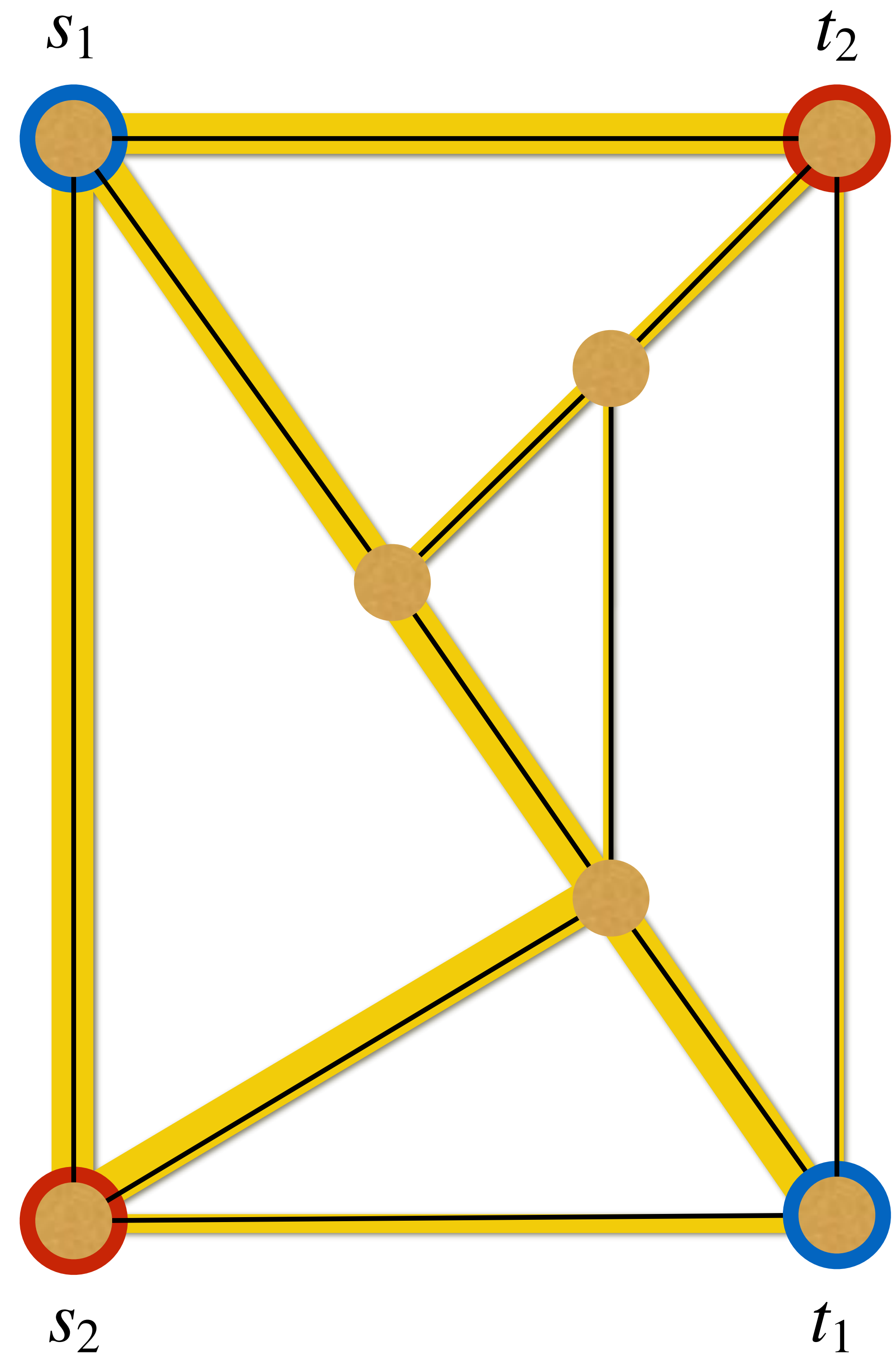
- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \to \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in \mathrm{V} \times \mathrm{V} \times \mathbb{R}_+$
  - construction costs $k_e \in \mathbb{R}_+$

Continuous network design problem

$$\min_{\mathbf{f},\, \mathbf{z}} \sum_{e \in E} \left(c_e(f_e / z_e) f_e + z_e k_e \right)$$

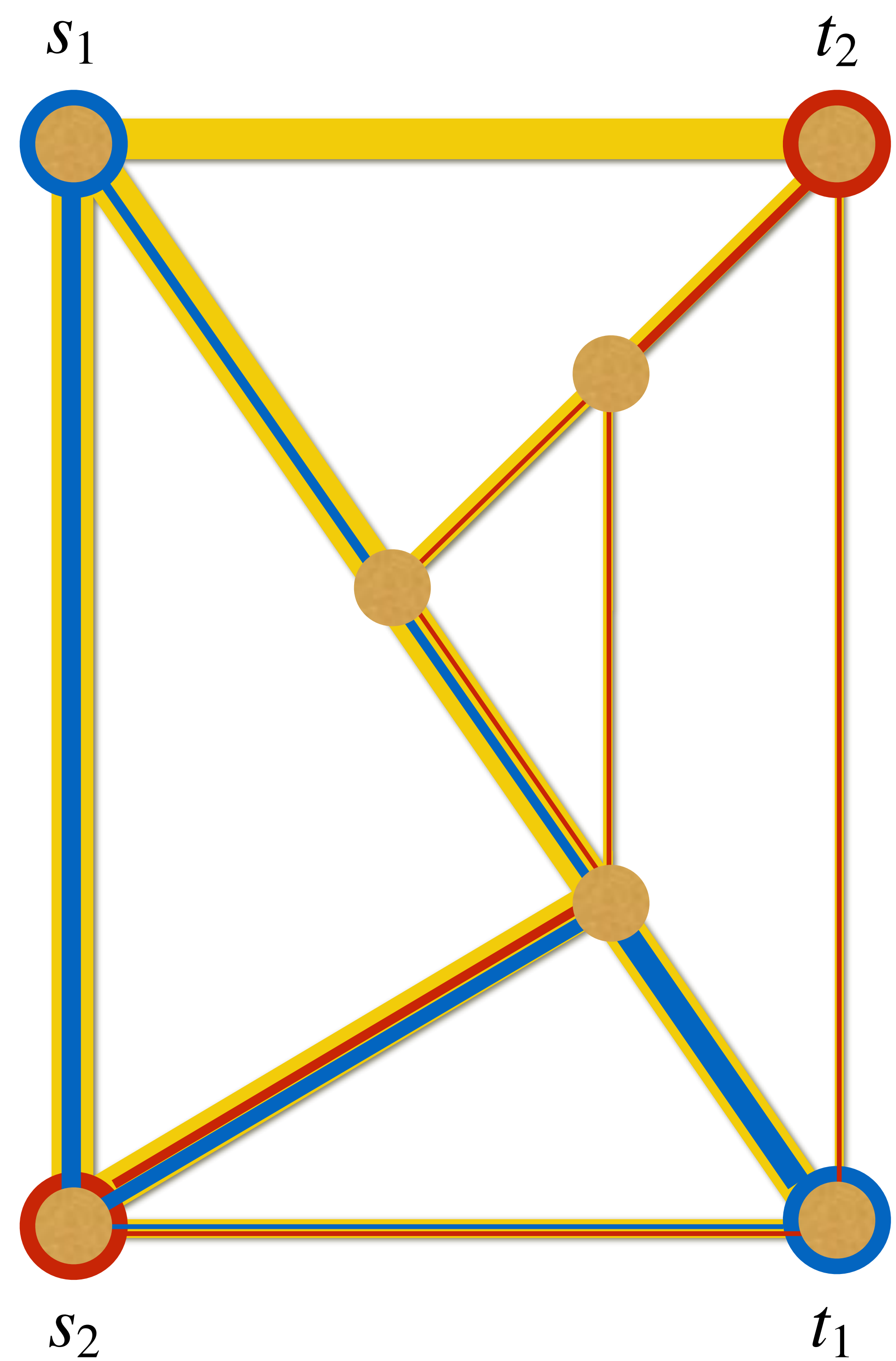
s.t.: $\mathbf{z} = (z_e)_{e \in E}$, $z_e \in \mathbb{R}_+$

$\mathbf{f} = (f_e)_{e \in E}$ is a Wardrop equilibrium



Continuous network design problem

- Graph $G = (V, E)$
  - edge cost functions $c_e : \mathbb{R}_+ \to \mathbb{R}_+$
    (strictly increasing, semi-convex, differentiable)
  - commodities $(s_i, t_i, d_i) \in V \times V \times \mathbb{R}_+$
  - construction costs $k_e \in \mathbb{R}_+$



Continuous network design problem

$$\min_{\mathbf{f},\, \mathbf{z}} \sum_{e \in E} \left(c_e(f_e / z_e) f_e + z_e k_e\right)$$

s.t.: $\mathbf{z} = (z_e)_{e \in E}$, $z_e \in \mathbb{R}_+$

$\mathbf{f} = (f_e)_{e \in E}$ is a Wardrop equilibrium

Continuous network design problem

$$\min_{\mathbf{f},\,\mathbf{z}} \sum_{e\in E} \left(c_e(f_e / z_e) f_e + z_e k_e\right)$$

$$\text{s.t.: } \mathbf{z} = (z_e)_{e\in E}\,,\, z_e \in \mathbb{R}_+$$

$\mathbf{f} = (f_e)_{e\in E}$ is a Wardrop equilibrium

Continuous network design problem

Continuous network design problem

$$\min_{\mathbf{f},\,\mathbf{z}} \sum_{e\in E} \left(c_e(f_e/\,z_e)f_e + z_e k_e\right)$$

s.t.: $\mathbf{z} = (z_e)_{e\in E}\,,\, z_e \in \mathbb{R}_+$

$\mathbf{f} = (f_e)_{e\in E}$ is a Wardrop equilibrium

equilibrium constraints

Continuous network design problem

Continuous network design problem

$\min_{\mathbf{f},\,\mathbf{z}} \sum_{e\in E} \left(c_e(f_e / z_e)f_e + z_e k_e\right)$

s.t.: $\mathbf{z} = (z_e)_{e\in E}\,,\, z_e \in \mathbb{R}_+$

$\mathbf{f} = (f_e)_{e\in E}$ is a Wardrop equilibrium — equilibrium constraints

$\Leftrightarrow$

$\mathbf{f} \in \operatorname{argmin}_{g\,:\,g\text{ flow}} \sum_{e\in E} \int_0^{g_e} c_e(t/z_e)\,\mathrm{d}t$ — potential function

Continuous network design problem

Continuous network design problem

$$\min_{\mathbf{f},\,\mathbf{z}} \sum_{e\in E} \left(c_e(f_e / z_e) f_e + z_e k_e\right)$$

s.t.: $\mathbf{z} = (z_e)_{e\in E}\,,\ z_e \in \mathbb{R}_+$

$\mathbf{f} = (f_e)_{e\in E}$ is a Wardrop equilibrium — equilibrium constraints

$\Leftrightarrow$

$\mathbf{f} \in \operatorname{argmin}_{\mathbf{g}\,:\,\mathbf{g}\text{ flow}} \sum_{e\in E} \int_0^{g_e} c_e(t/z_e)\,\mathrm{d}t$ — potential function

$\Leftrightarrow$

$\sum_{e\in E} c_e(f_e/z_e)(f_e - g_e) \leq 0$ for all flows $\mathbf{g}$ — variational inequality

Example

flow f_e

capacity z_e

latency $c_e(f_e/z_e)$

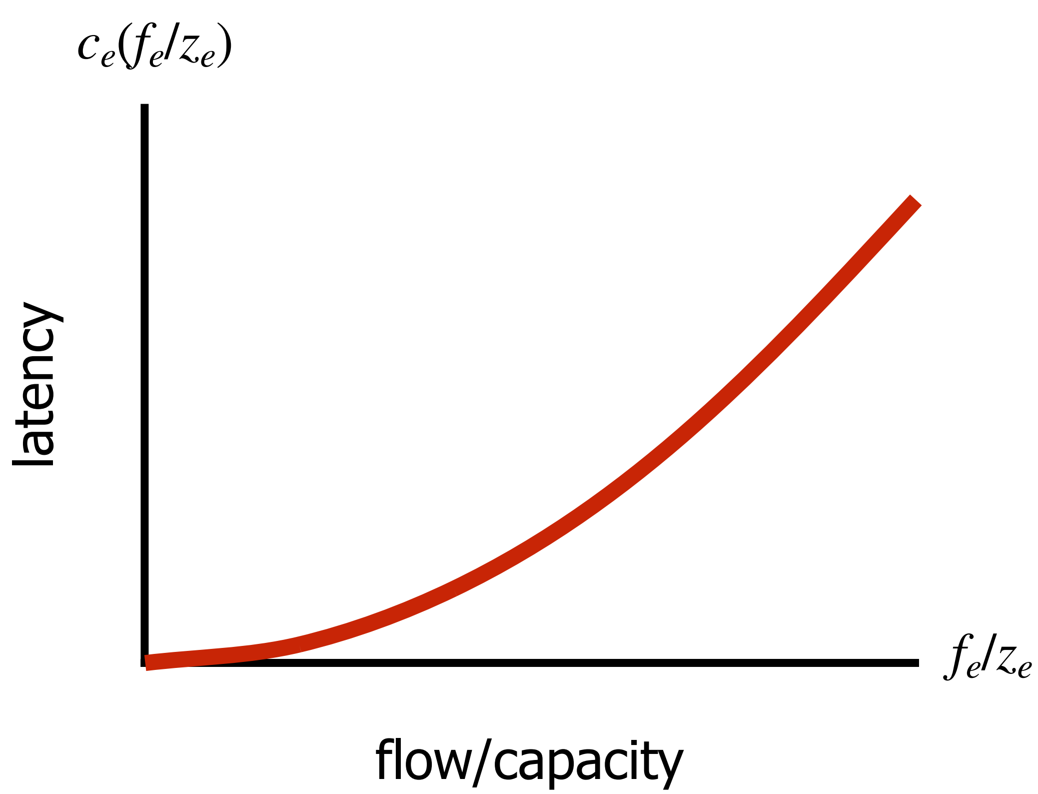
Example

Example

flow f_e capacity z_e latency $c_e(f_e/z_e)$

Example


road e
flow f_e
capacity z_e
latency $c_e(f_e/z_e)$

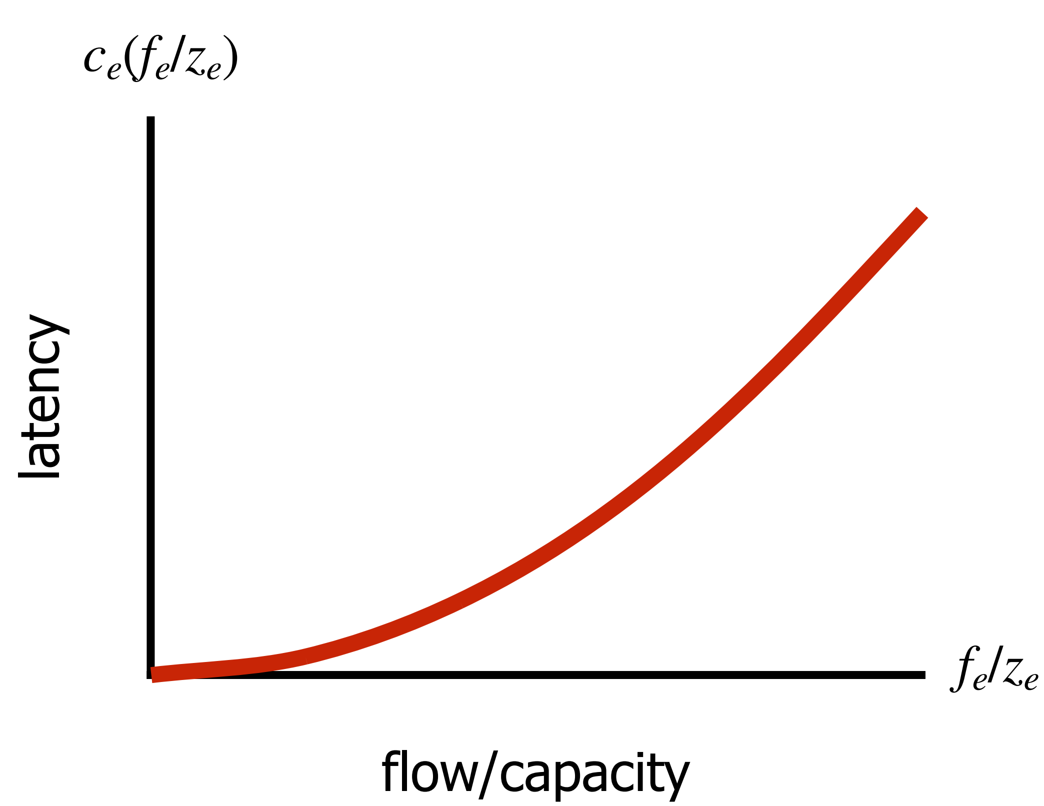

$c_e(f_e/z_e)$
latency
f_e/z_e
flow/capacity

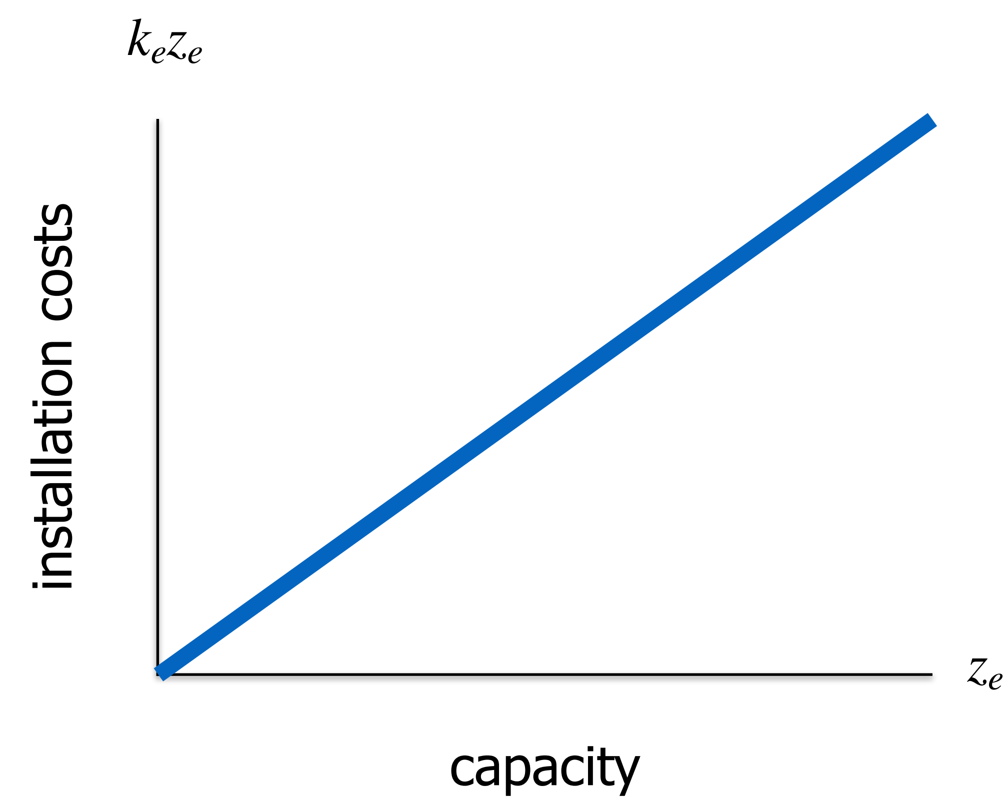
Example

flow f_e

capacity z_e

latency $c_e(f_e/z_e)$





Example

- optimal capacity for $c_e(f_e/z_e) = f_e/z_e + b$

Example

flow f_e capacity z_e latency $c_e(f_e/z_e)$

- optimal capacity for $c_e(f_e/z_e) = f_e/z_e + b$
  - total costs for e: C_e = routing costs + installation costs

$$= c_e(f_e/z_e)f_e + k_e z_e = f_e^2/z_e + bf_e + k_e z_e$$

Example

- optimal capacity for $c_e(f_e/z_e) = f_e/z_e + b$
  - total costs for e: C_e = routing costs + installation costs
    $= c_e(f_e/z_e)f_e + k_e z_e = f_e^2/z_e + bf_e + k_e z_e$
  - optimal capacity for e: $z_e^* = f_e/\sqrt{k_e}$

Example

flow f_e capacity z_e latency $c_e(f_e/z_e)$

- optimal capacity for $c_e(f_e/z_e) = f_e/z_e + b$
  - total costs for e: C_e = routing costs + installation costs
    $$= c_e(f_e/z_e)f_e + k_e z_e = f_e^2/z_e + bf_e + k_e z_e$$
  - optimal capacity for e: $z_e^* = f_e/\sqrt{k_e}$
  - optimal latency for e: $c_e^* = \sqrt{k_e} + b$

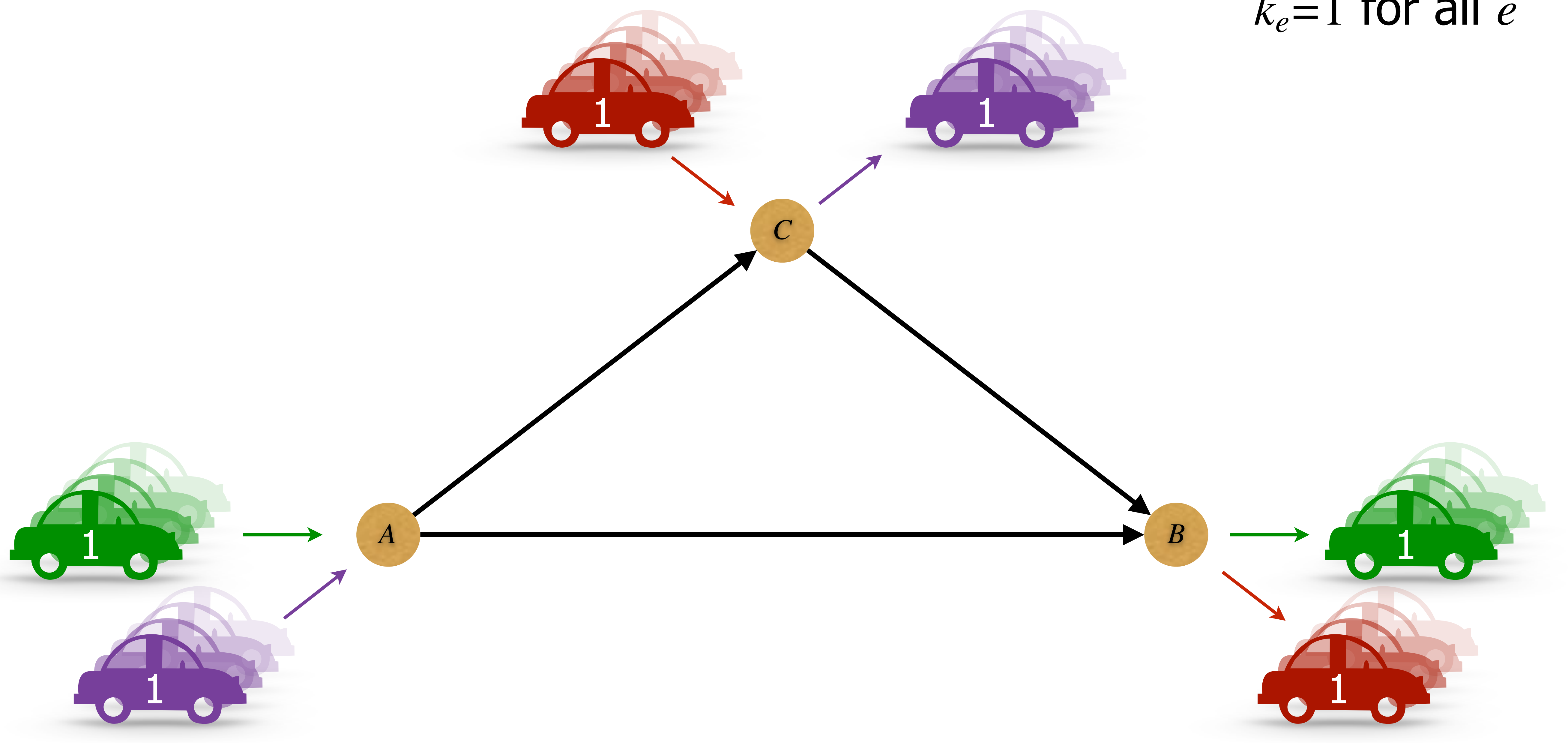
Example

flow f_e capacity z_e latency $c_e(f_e/z_e)$

- optimal capacity for $c_e(f_e/z_e) = f_e/z_e + b$
  - total costs for e: $C_e = \text{routing costs} + \text{installation costs}$
    $= c_e(f_e/z_e)f_e + k_e z_e = f_e^2/z_e + bf_e + k_e z_e$
  - optimal capacity for e: $z_e^* = f_e/\sqrt{k_e}$
  - optimal latency for e: $c_e^* = \sqrt{k_e} + b$
  - optimal total costs for e: $C_e^* = (\sqrt{k_e} + b + \sqrt{k_e})\, f_e$

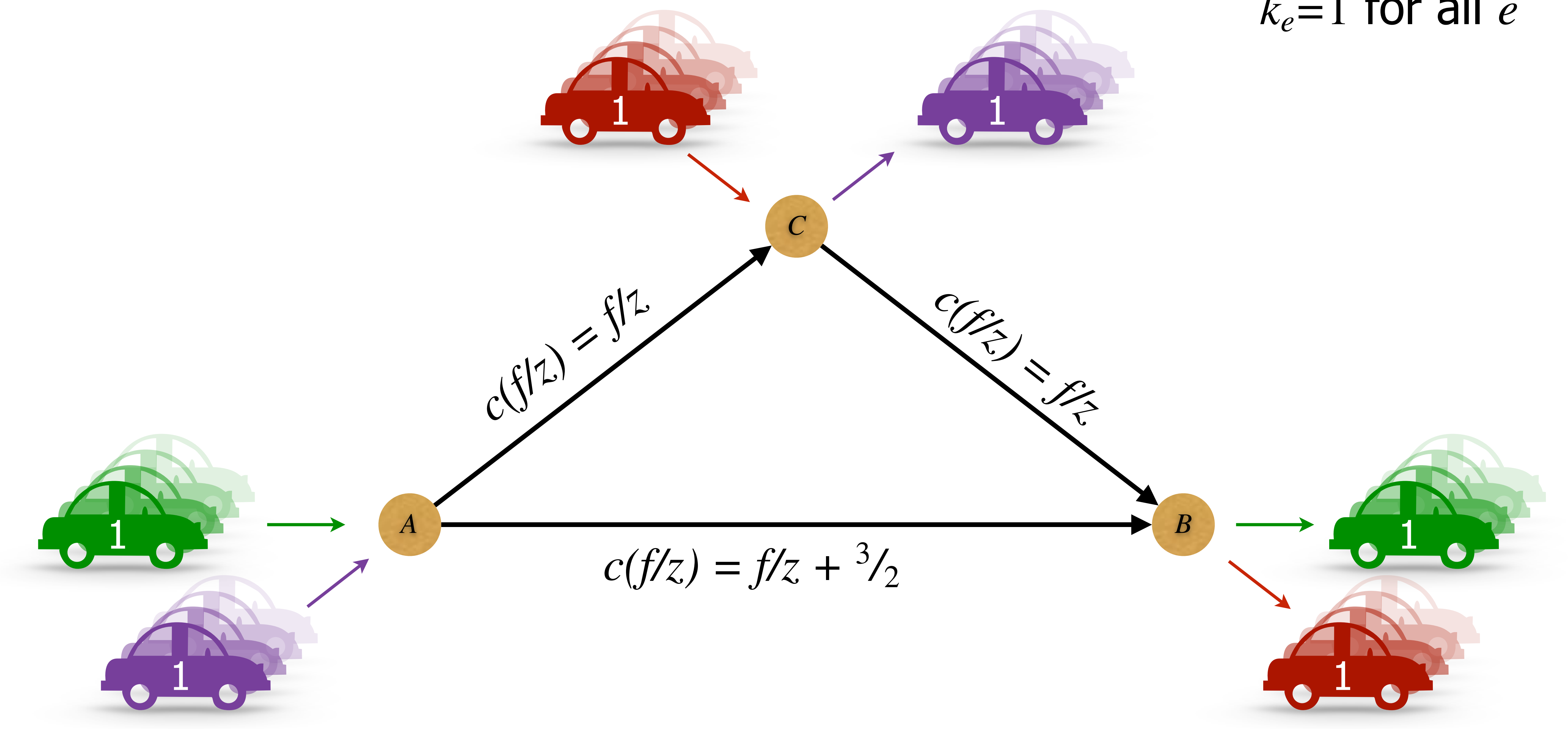
Example

$k_e = 1$ for all e



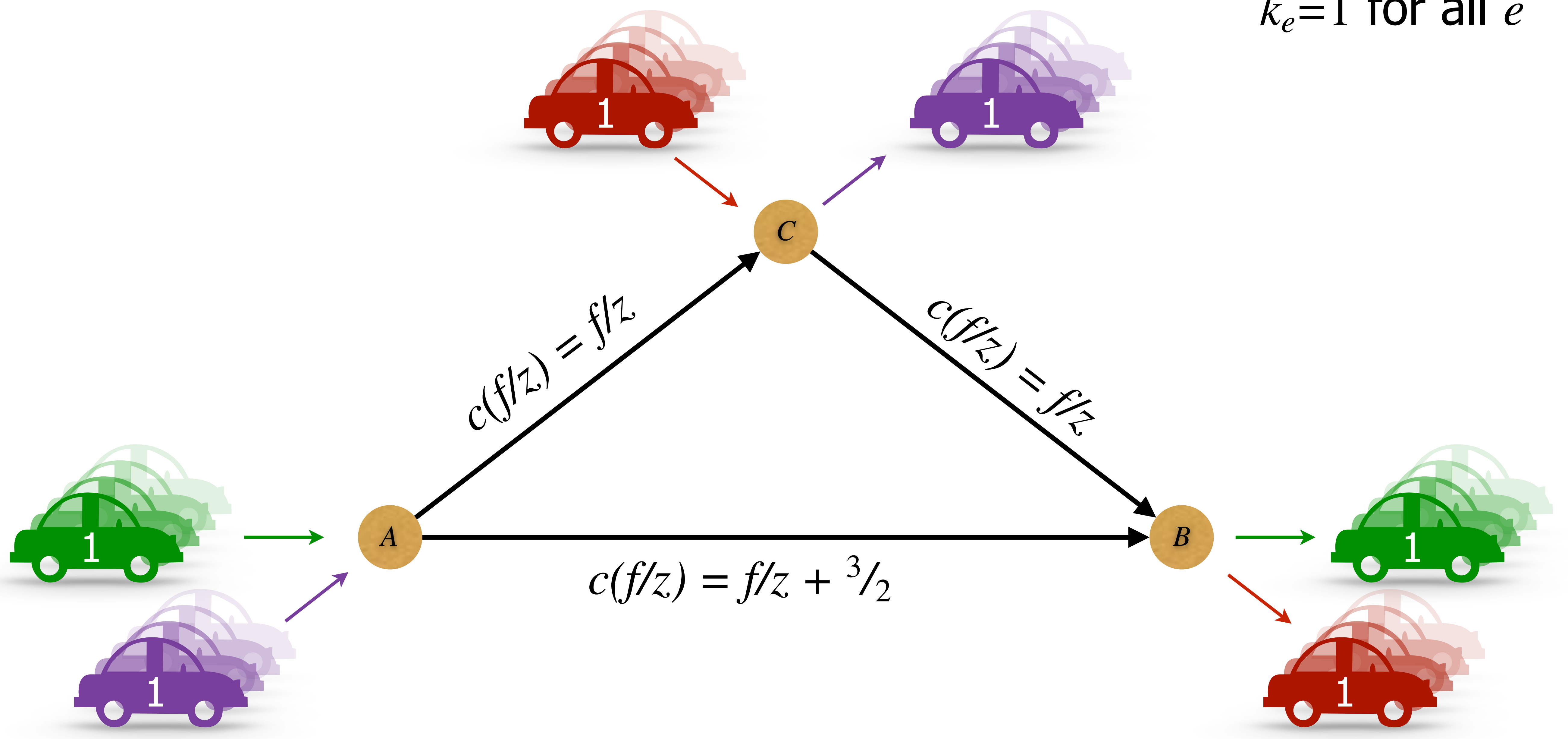
Example

k_e=1 for all e



Example

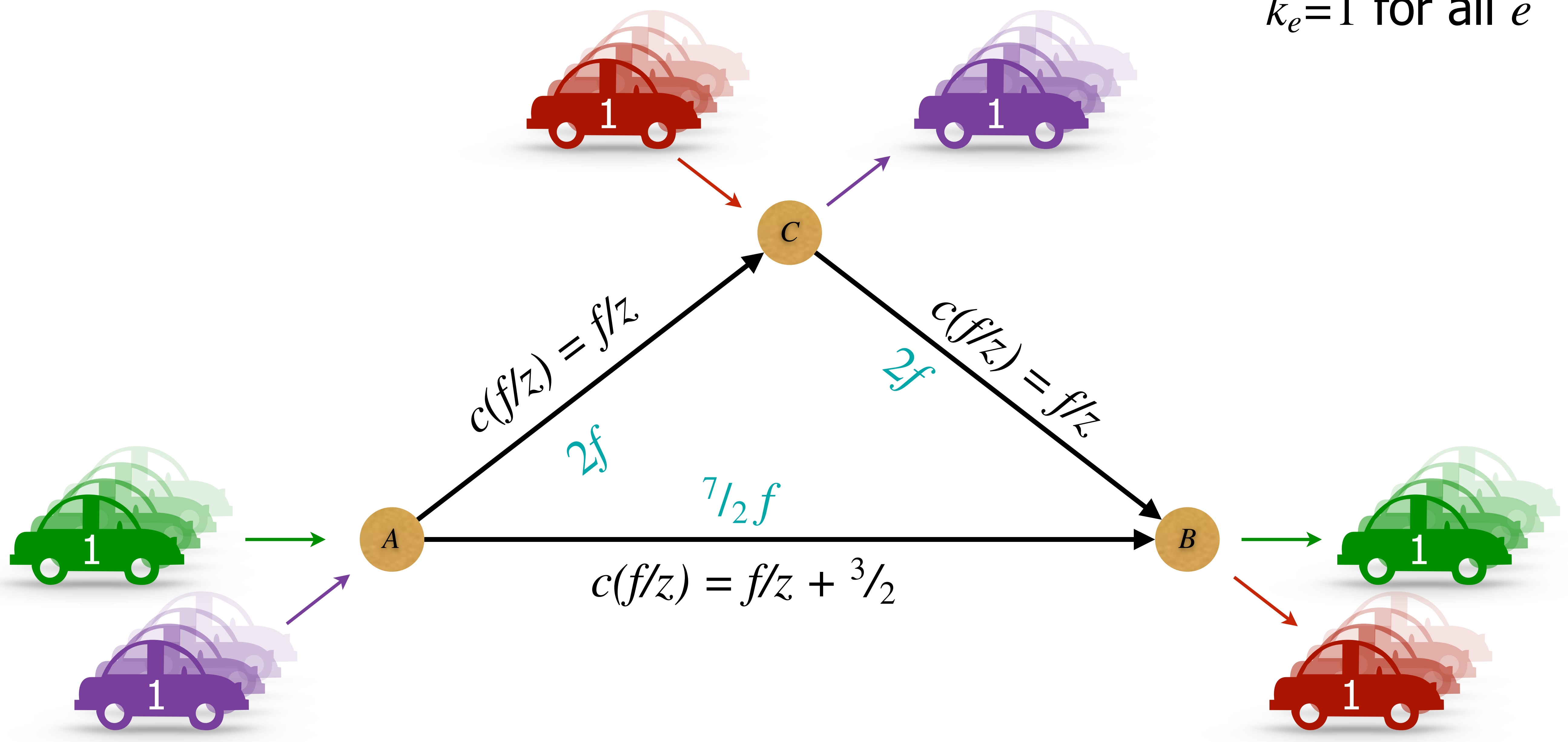
$k_e = 1$ for all e



- $\min_z\ (f/z+b)f + z$ attained for $z^* = f$

Example

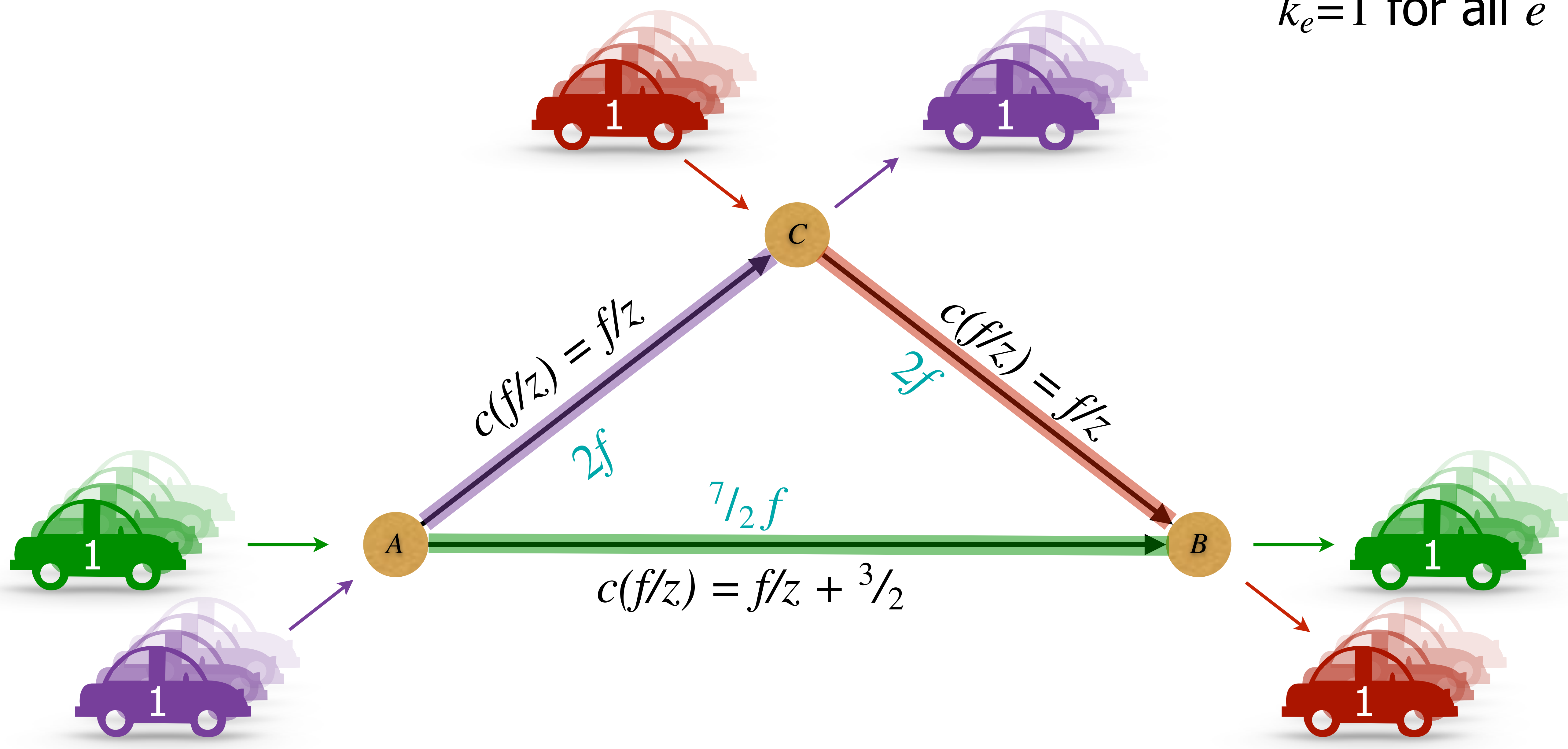
$k_e = 1$ for all e



- $\min_z\ (f/z + b)f + z$ attained for $z^* = f$
- opt. total cost: $(2 + b)f$

Example

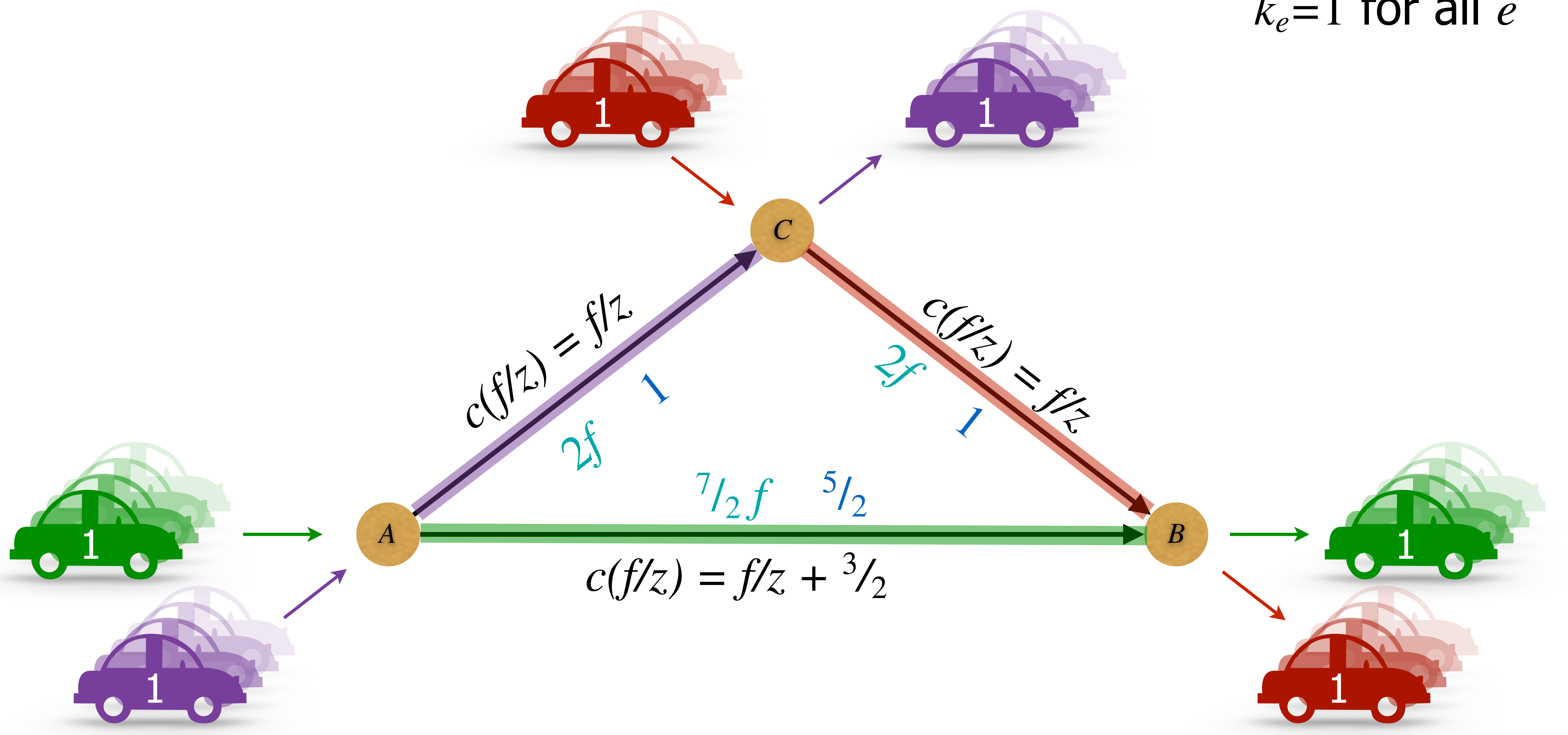
$k_e = 1$ for all e



- $\min_z \, (f/z+b)f + z$ attained for $z^* = f$
- opt. total cost: $(2 + b)f$

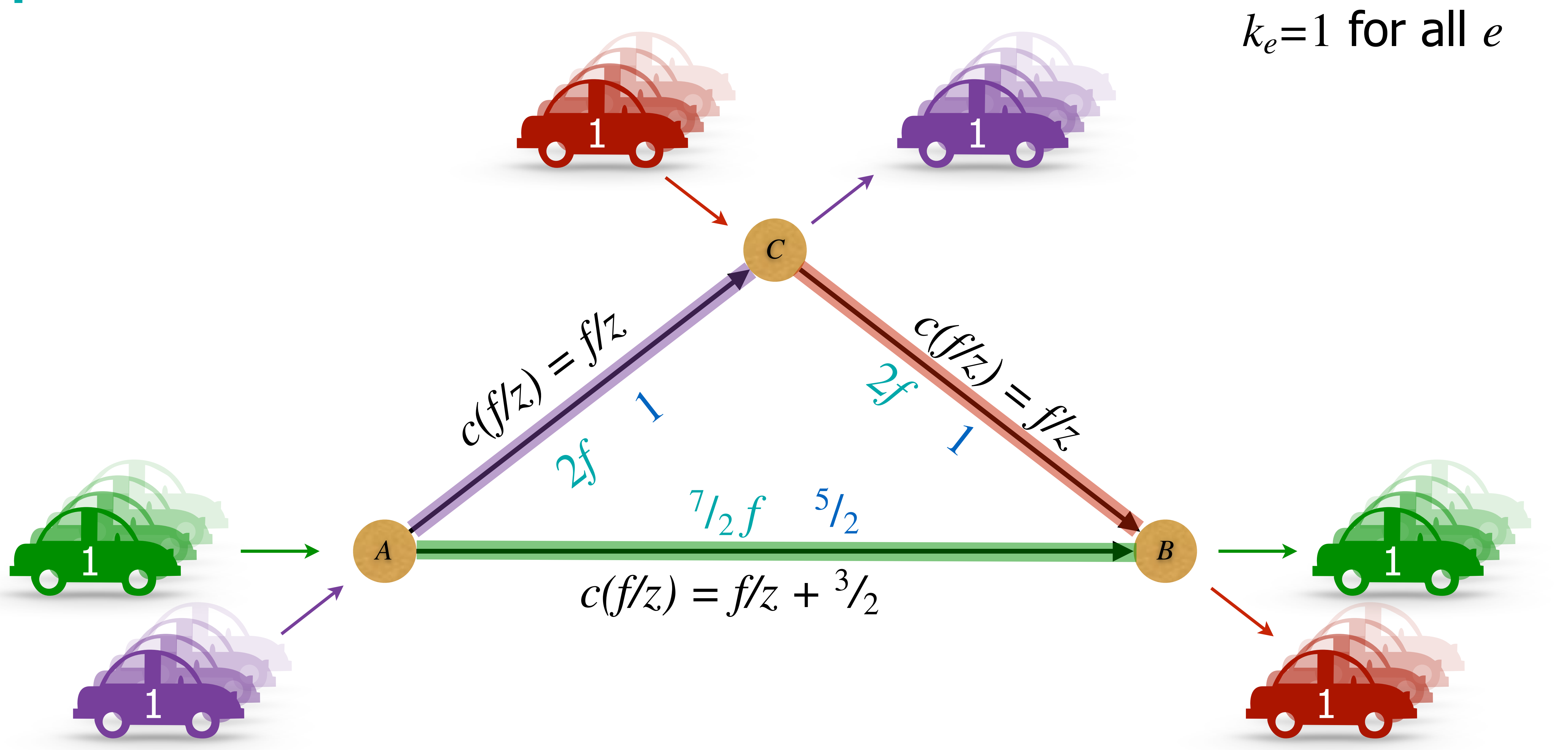
Example

$k_e = 1$ for all e



- $\min_z\ (f/z+b)f + z$ attained for $z^* = f$
- opt. total cost: $(2 + b)f$
- experienced latency: $1 + b$

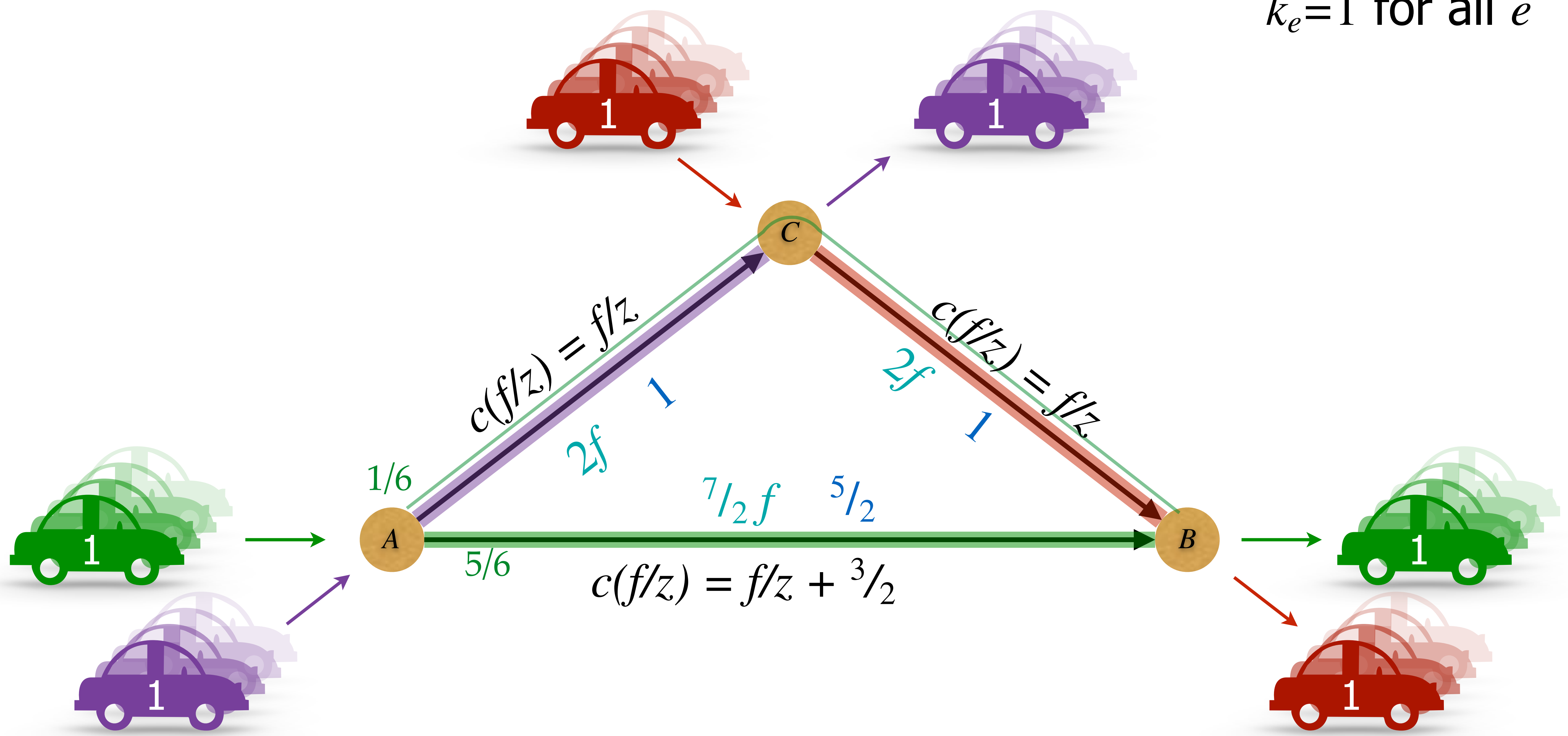
Example



- $\min_z\ (f/z + b)f + z$ attained for $z^* = f$
- opt w/o equilibrium: $7 + {}^1/_2$
- opt. total cost: $(2 + b)f$
- experienced latency: $1 + b$

Example

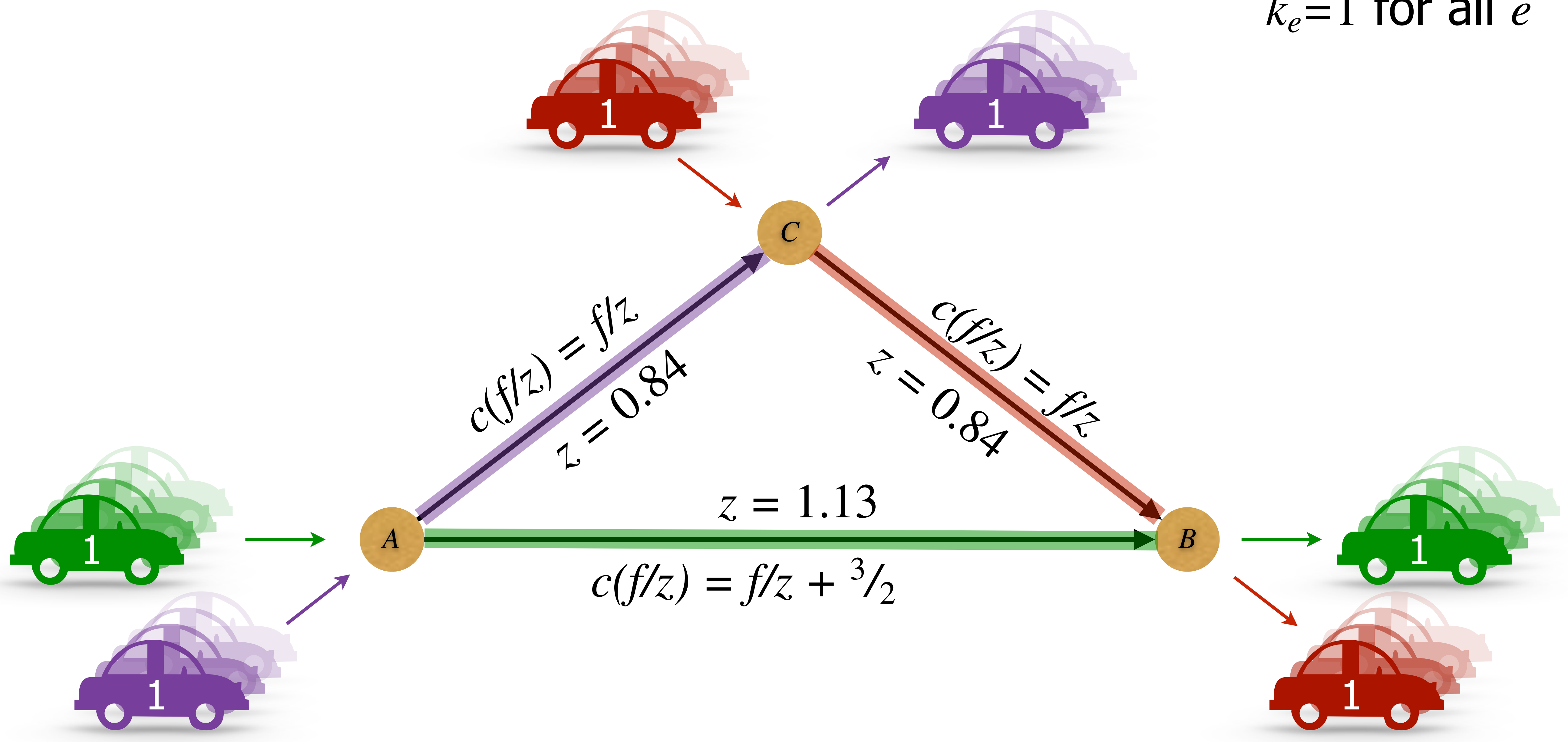
$k_e = 1$ for all e



- $\min_z (f/z+b)f + z$ attained for $z^* = f$
- opt. total cost: $(2 + b)f$
- experienced latency: $1 + b$
- opt w/o equilibrium: $7 + {}^1/_2$
- sol w/ equilibrium: $7 + {}^2/_3$

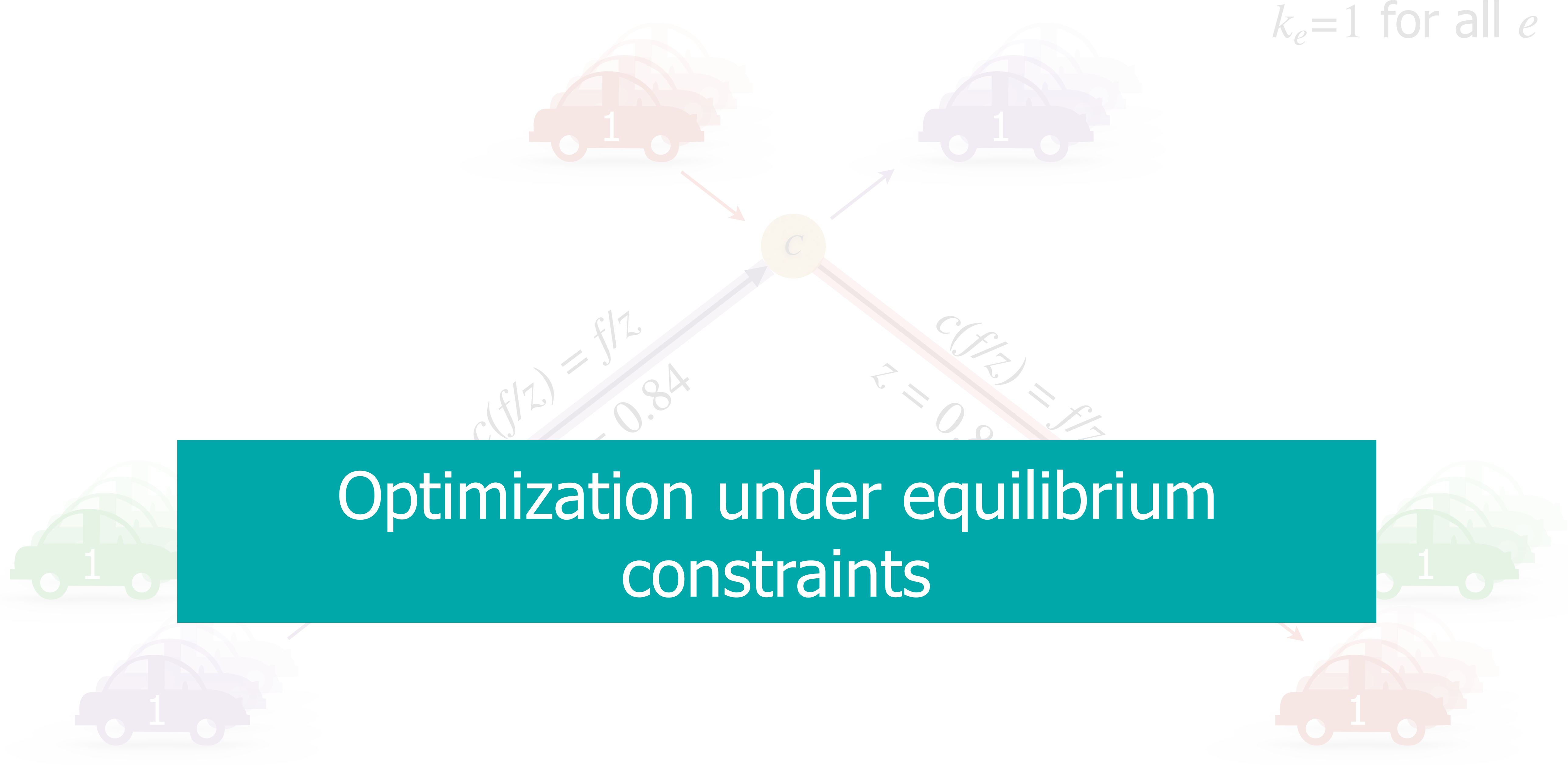
Example

$k_e = 1$ for all e



- $\min_z \, (f/z + b)f + z$ attained for $z^* = f$
- opt. total cost: $(2 + b)f$
- experienced latency: $1 + b$
- opt w/o equilibrium: $7 + {}^{1}/_{2}$
- sol w/ equilibrium: $7 + {}^{2}/_{3}$
- opt w/ equilibrium: 7.577

Example



Optimization under equilibrium constraints

- $\min_z (f/z+b)f + z$ attained for $z^* = f$
- opt. total cost: $(2 + b)f$
- experienced latency: $1 + b$
- opt w/o equilibrium: $7 + {}^1/_2$
- sol w/ equilibrium: $7 + {}^2/_3$
- opt w/ equilibrium: 7.577

Previous work

- "one of the most important, difficult and challenging problems in transport" [Yang, Bell '98]
- various heuristics e.g. [Dafermos '69][Dantzig et al. '79]
- approximation algorithm [Marcotte '85]
  - $5/4$-approximation for affine costs $c(f/z) = a + b(f/z)$
  - closed formula for monomials $c(f/z) = a + b(f/z)^d$, converges to 2 as $d \to \infty$

Previous work   Our contribution

- "one of the most important, difficult and challenging problems in transport" → APX-hardness [Yang, Bell '98]

- various heuristics e.g. [Dafermos '69][Dantzig et al. '79]

- approximation algorithm [Marcotte '85]
  - 5/4-approximation for affine costs $c(f/z) = a + b(f/z)$
  - closed formula for monomials $c(f/z) = a + b(f/z)^d$, converges to 2 as $d \to \infty$

→ approximation algorithm with improved guarantee

Complexity

Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

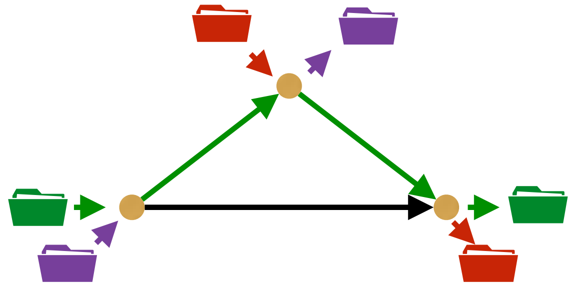
- even if all costs are affine.

Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.

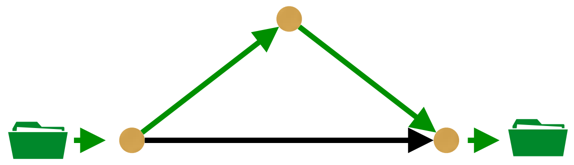


Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.

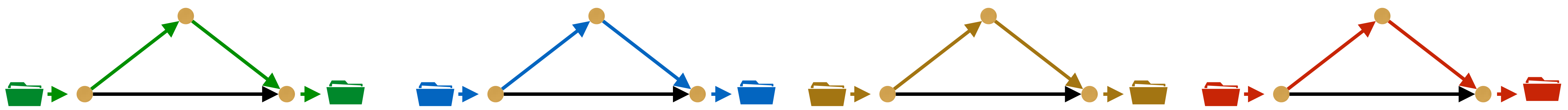


Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.

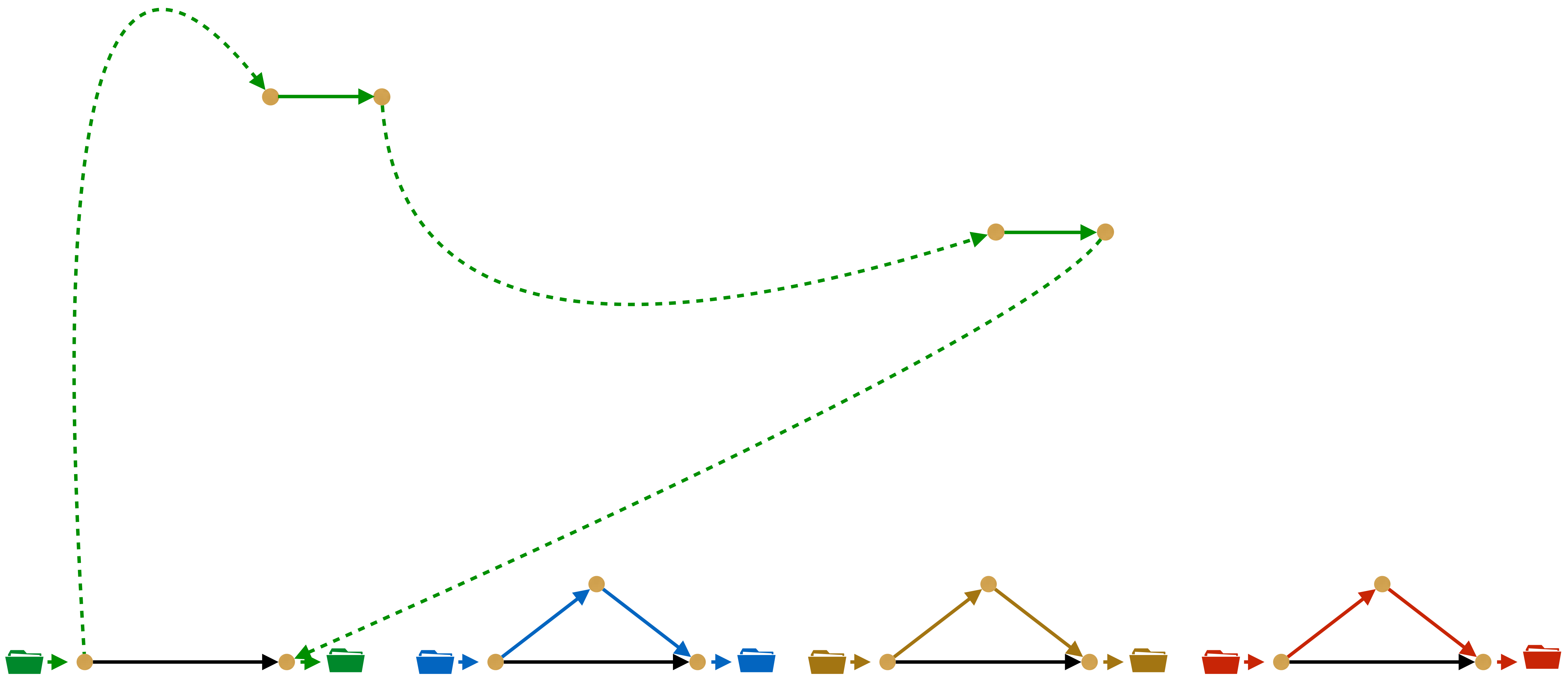


Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.



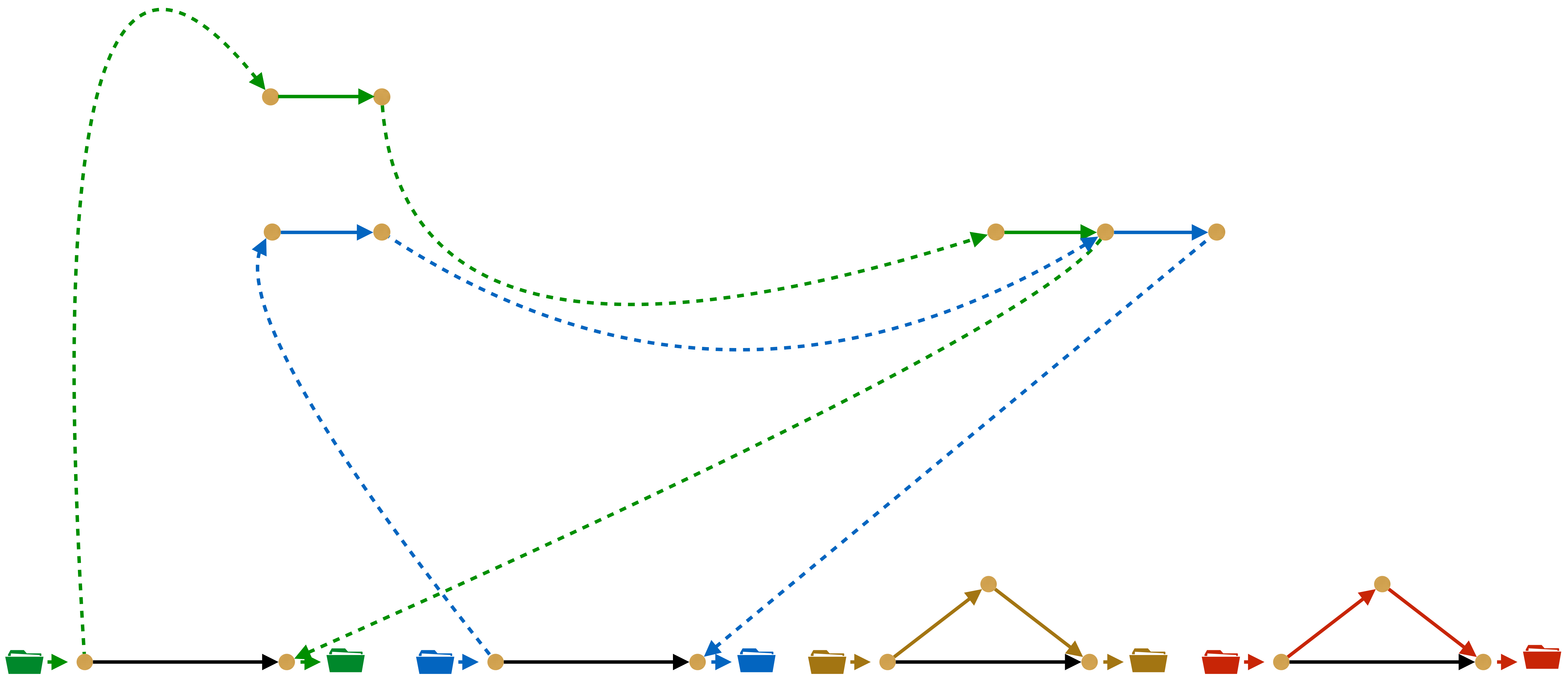
Complexity

Theorem

[Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.

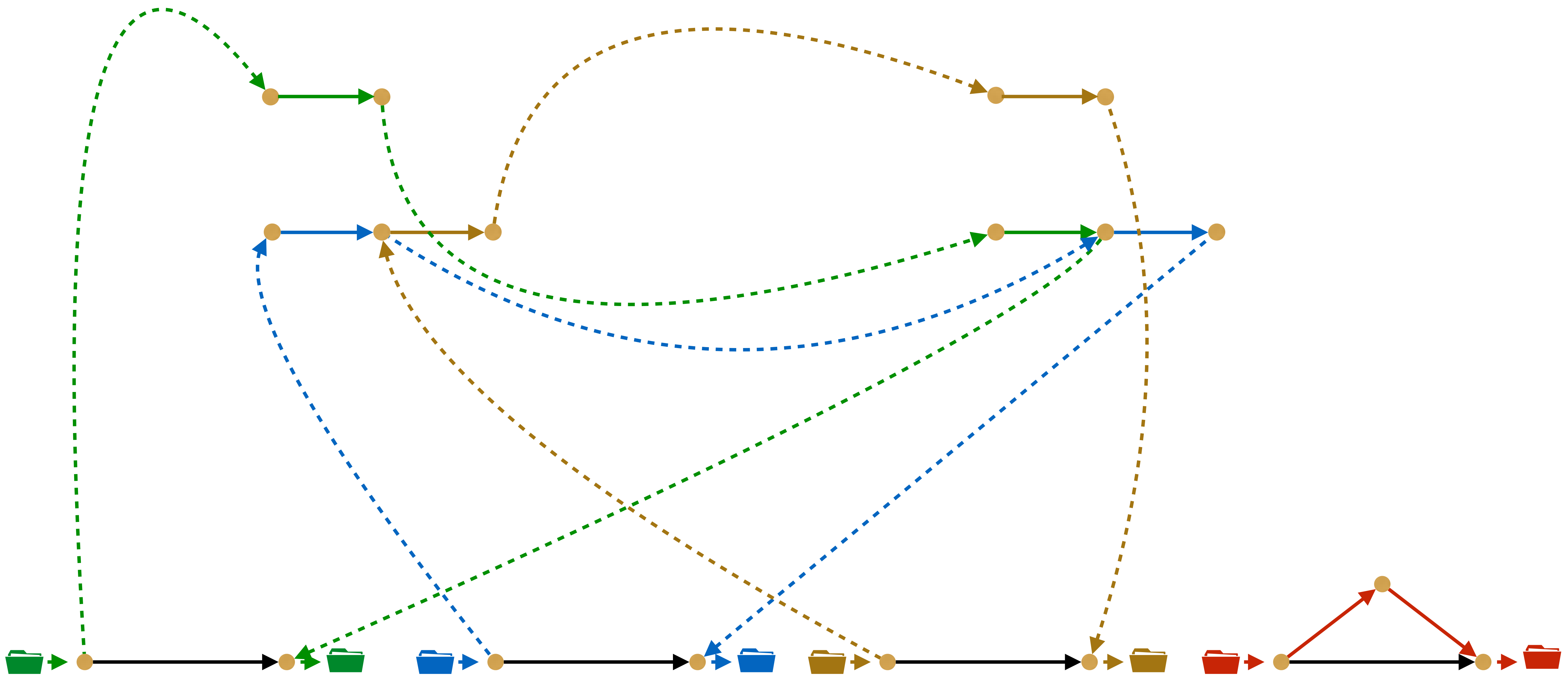


Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.

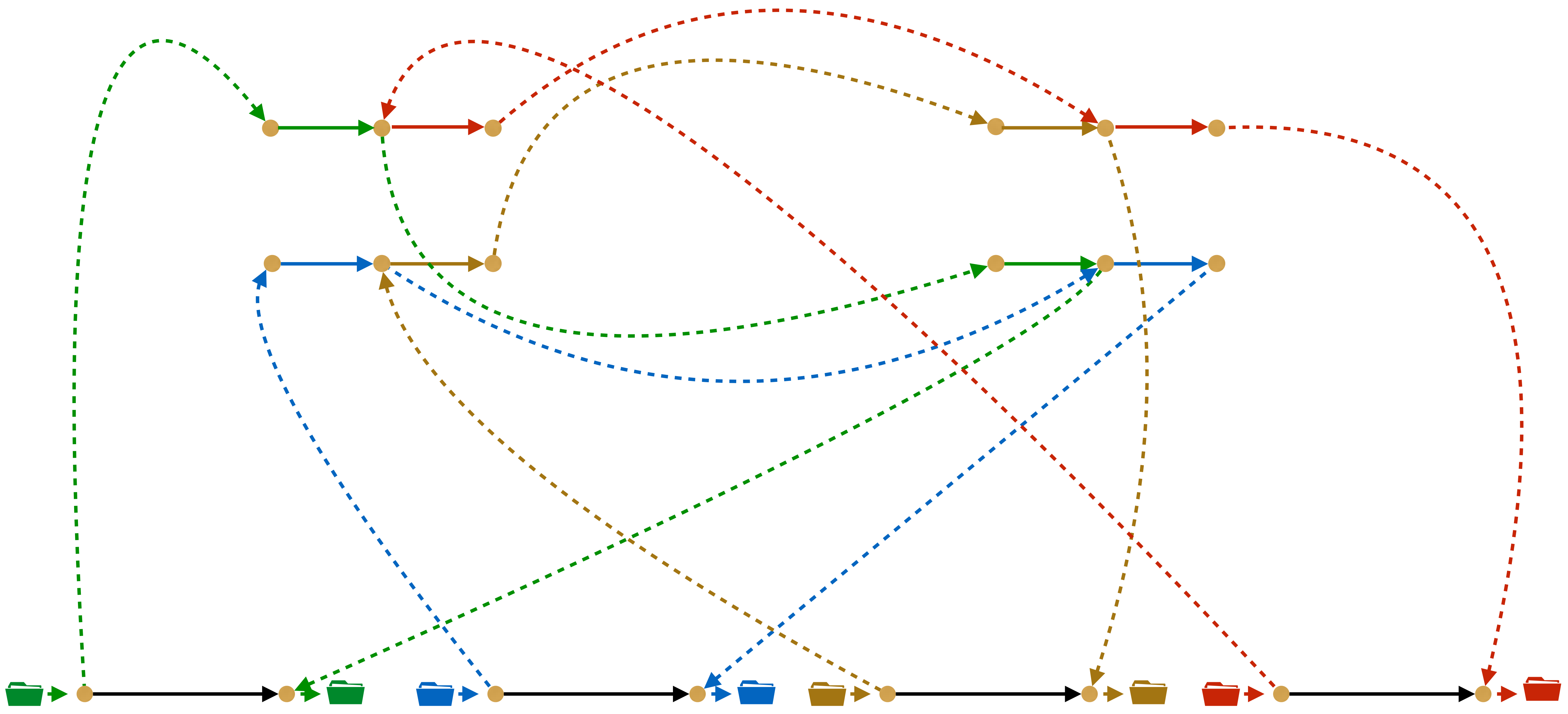


Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.



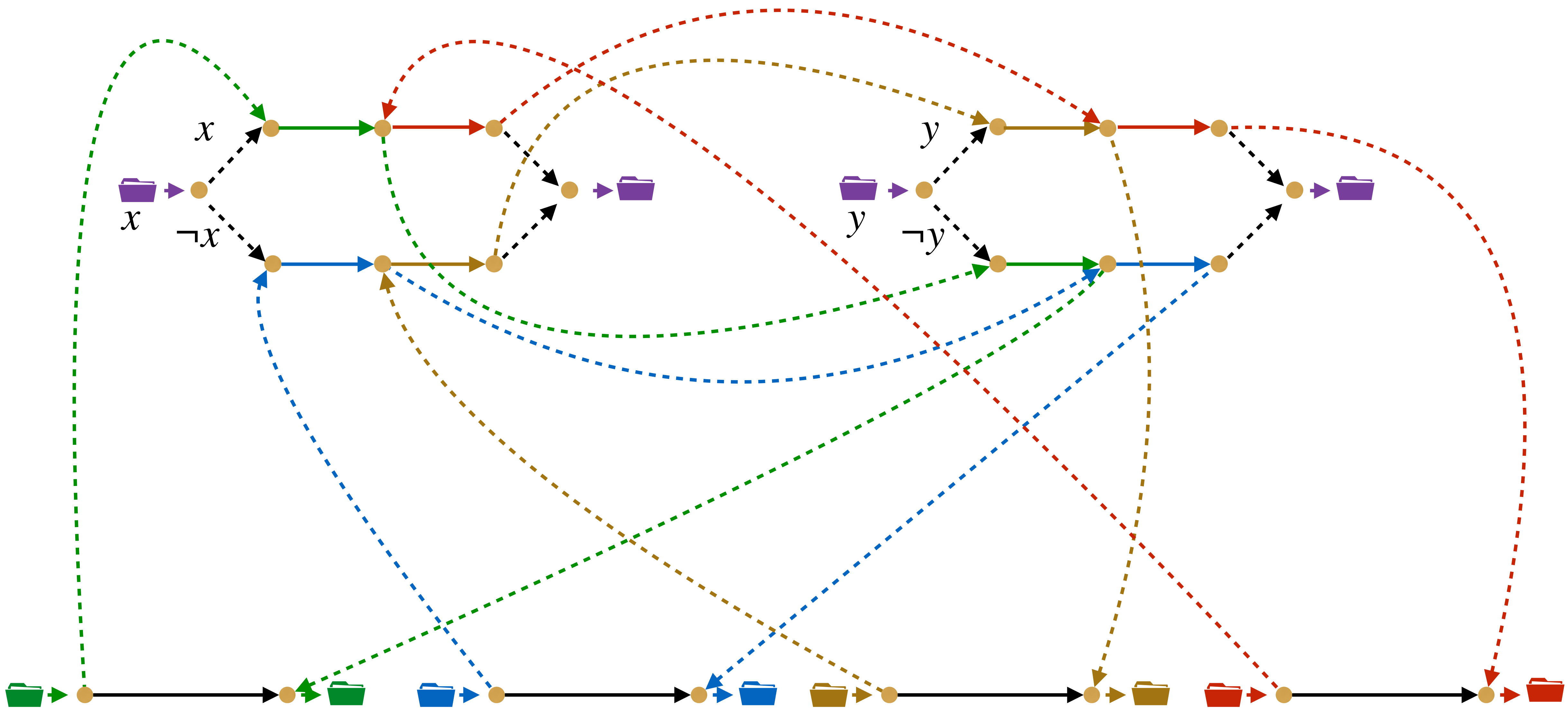
Complexity

Theorem

[Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.

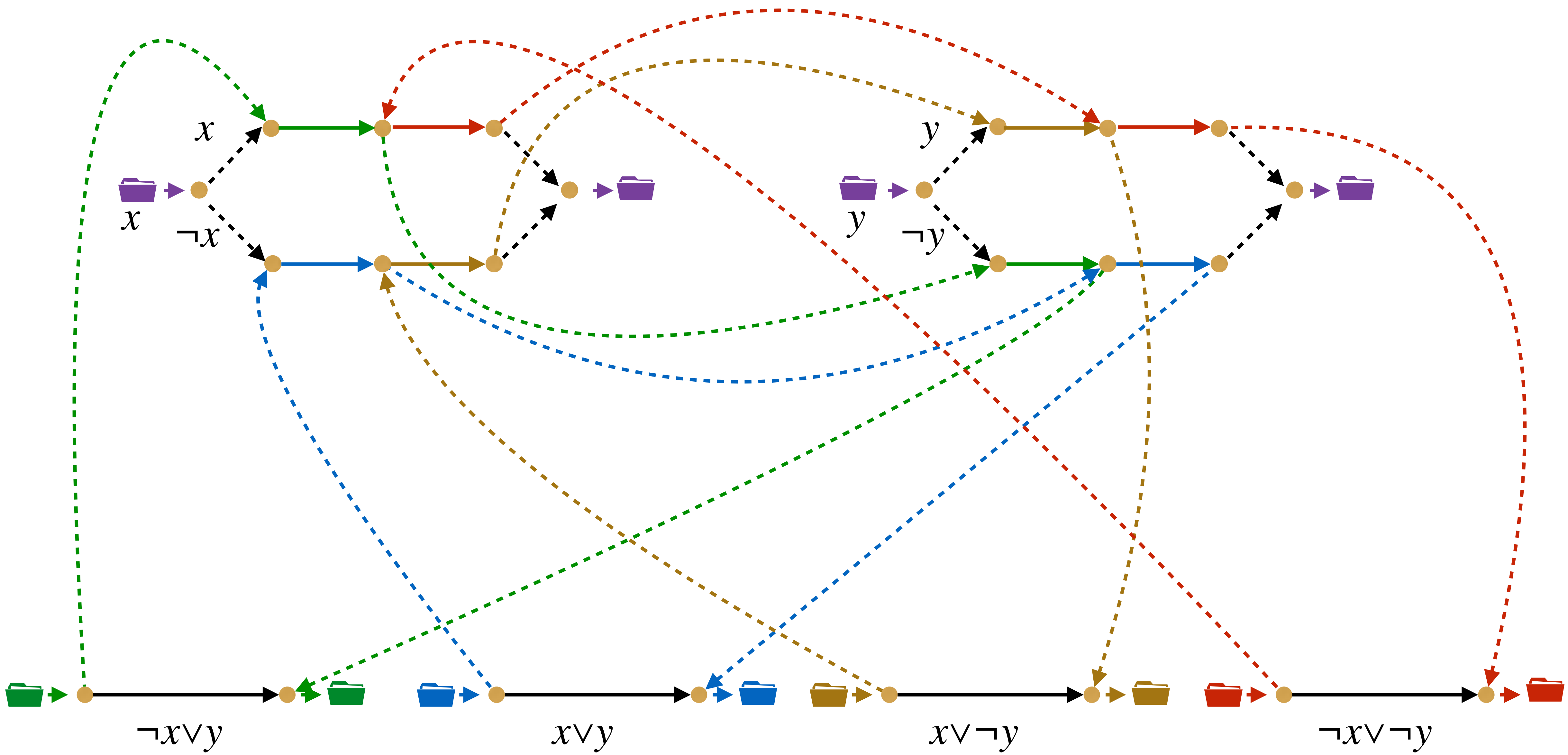


Complexity

Theorem [Gairing, Harks & K., '14]

The CNDP is **APX**-hard.

- even if all costs are affine.



Approximation

The relaxation

Continuous network design problem — Relaxation

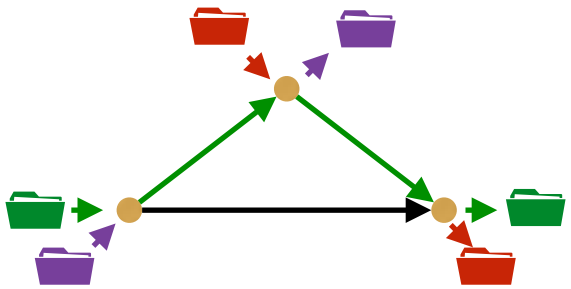
$$\min_{\mathbf{f},\, \mathbf{z}} \sum_{e \in E} \left(c_e(f_e / z_e) f_e + z_e k_e \right)$$

$$\text{s.t.: } \mathbf{z} = (z_e)_{e \in E},\, z_e \in \mathbb{R}_+$$

$$\mathbf{f} = (f_e)_{e \in E} \text{ is a flow}$$

Lemma [Marcotte '85; Harks, Gairing, K., '14]

The relaxation can be solved in polynomial time.



Quality of the relaxation

- Price of Anarchy (PoA) = $\dfrac{\text{Routing Cost of Wardrop equilibrium}}{\text{Routing cost of optimum}}$

Theorem [Roughgarden, Tardos ’02; Correa at al. ‘04]

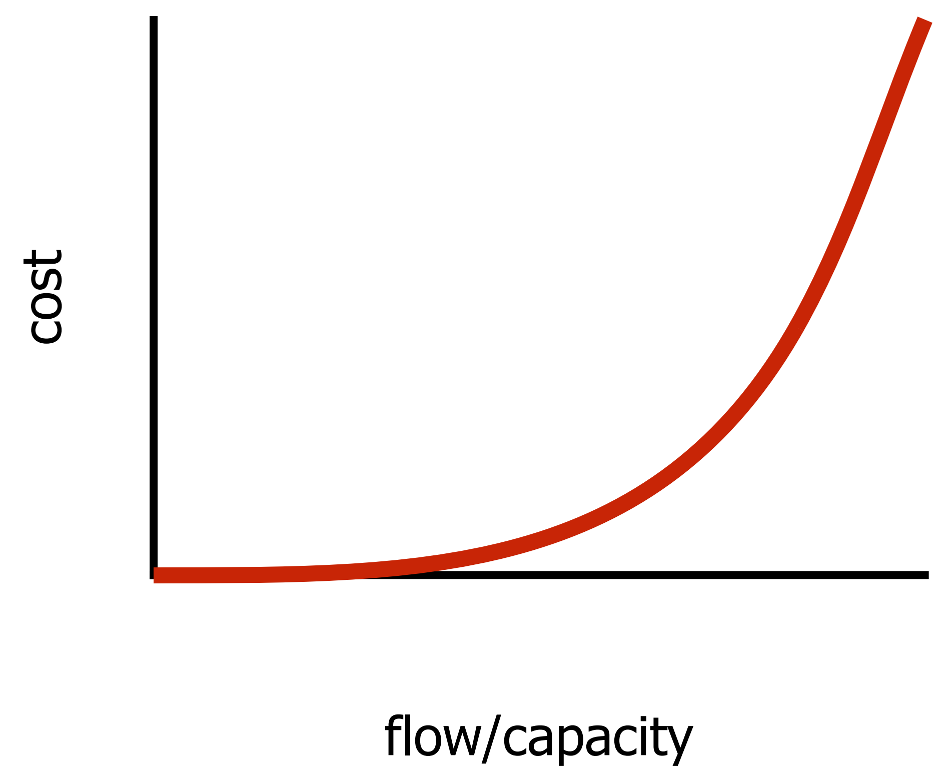
$\mathrm{PoA} \leq \frac{1}{1-\mu}$ where $\mu = \sup_{c \in \mathscr{C},\, x \geq 0} \max_{\gamma \in [0,1]} \gamma \left(1 - \frac{c(\gamma \cdot x)}{c(x)}\right).$

Quality of the relaxation

- Price of Anarchy (PoA) = $\dfrac{\text{Routing Cost of Wardrop equilibrium}}{\text{Routing cost of optimum}}$

Theorem [Roughgarden, Tardos '02; Correa at al. '04]

$\mathrm{PoA} \leq \frac{1}{1-\mu}$ where $\mu = \sup_{c \in \mathscr{C},\, x \geq 0} \max_{\gamma \in [0,1]} \gamma \left(1 - \frac{c(\gamma \cdot x)}{c(x)}\right).$

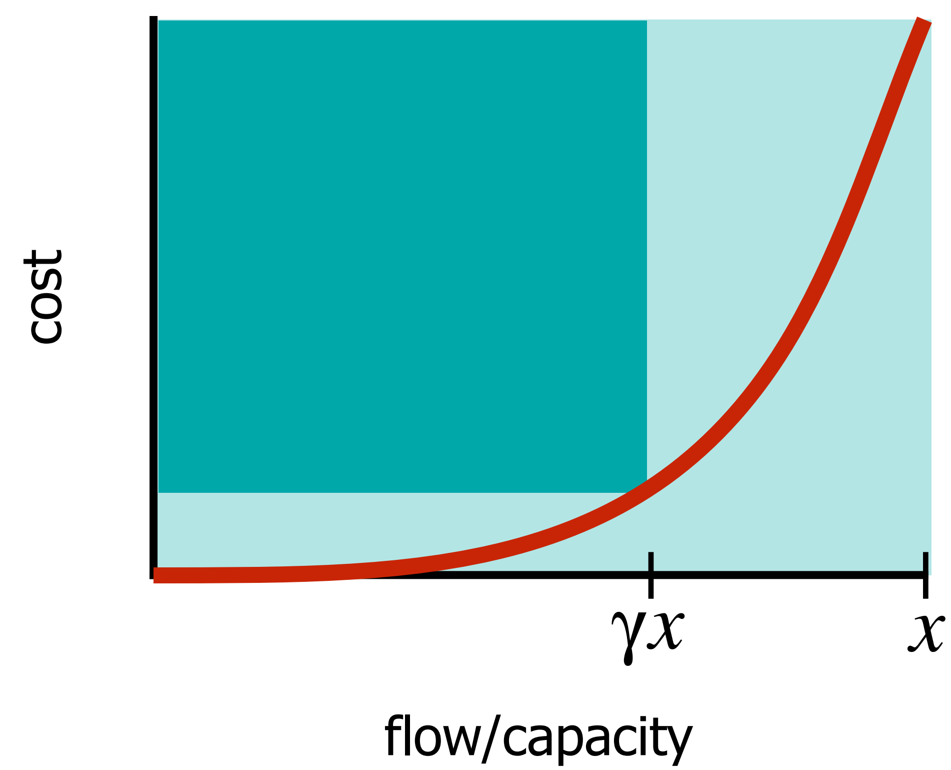


Quality of the relaxation

- Price of Anarchy (PoA) = $\frac{\text{Routing Cost of Wardrop equilibrium}}{\text{Routing cost of optimum}}$

Theorem [Roughgarden, Tardos '02; Correa at al. '04]

$\mathrm{PoA} \leq \frac{1}{1-\mu}$ where $\mu = \sup_{c \in \mathscr{C},\, x \geq 0} \max_{\gamma \in [0,1]} \gamma \left(1 - \frac{c(\gamma \cdot x)}{c(x)}\right)$.



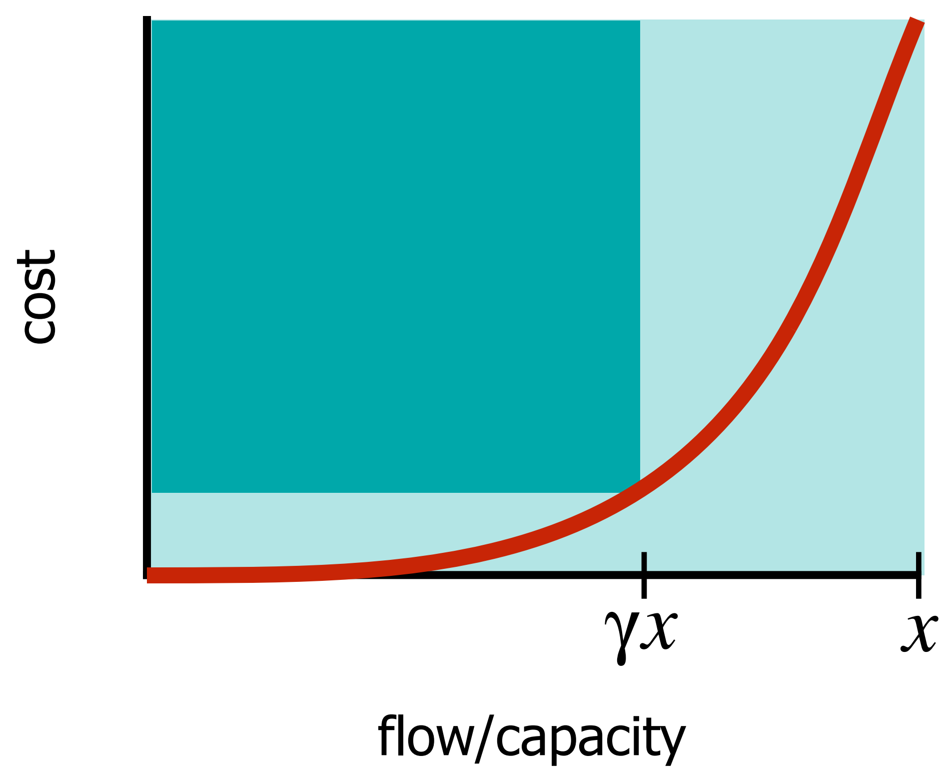
- μ = ■ / ■

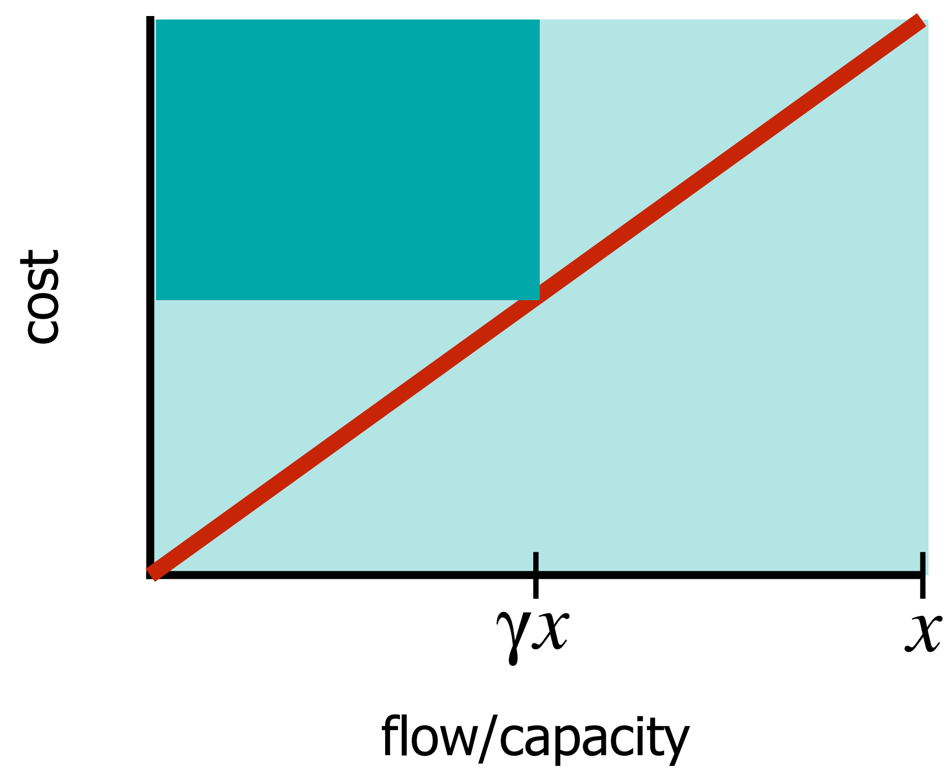
Quality of the relaxation

- Price of Anarchy (PoA) = $\dfrac{\text{Routing Cost of Wardrop equilibrium}}{\text{Routing cost of optimum}}$

Theorem [Roughgarden, Tardos '02; Correa at al. '04]

$\mathrm{PoA} \leq \frac{1}{1-\mu}$ where $\mu = \sup_{c \in \mathscr{C},\, x \geq 0} \max_{\gamma \in [0,1]} \gamma \left(1 - \frac{c(\gamma \cdot x)}{c(x)}\right)$.





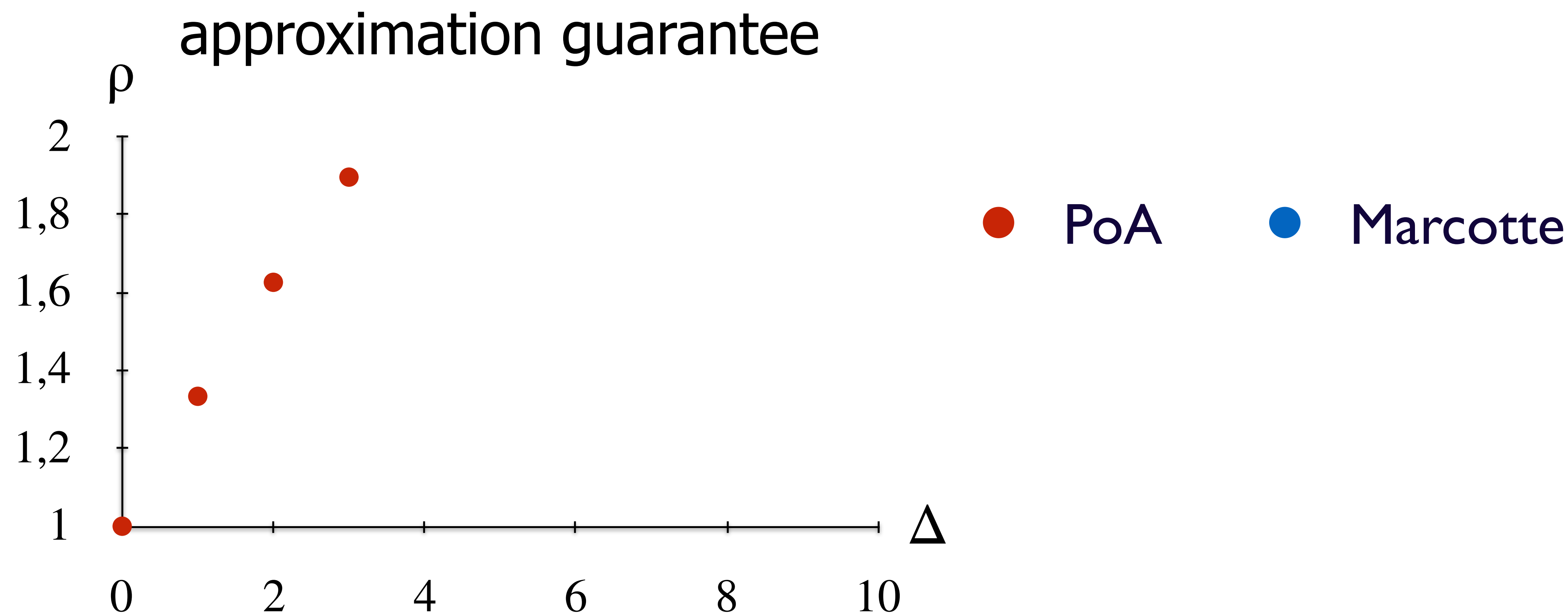
- $\mu =$ ■ / ■
- affine: $\mu = 1/4$ for $\gamma = 1/2$

Known results

- Solve the relaxation → Compute a Wardrop equilibrium
  - $1/(1-\mu(\mathscr{C}))$-approximation for all $\mathscr{C}$

Known results

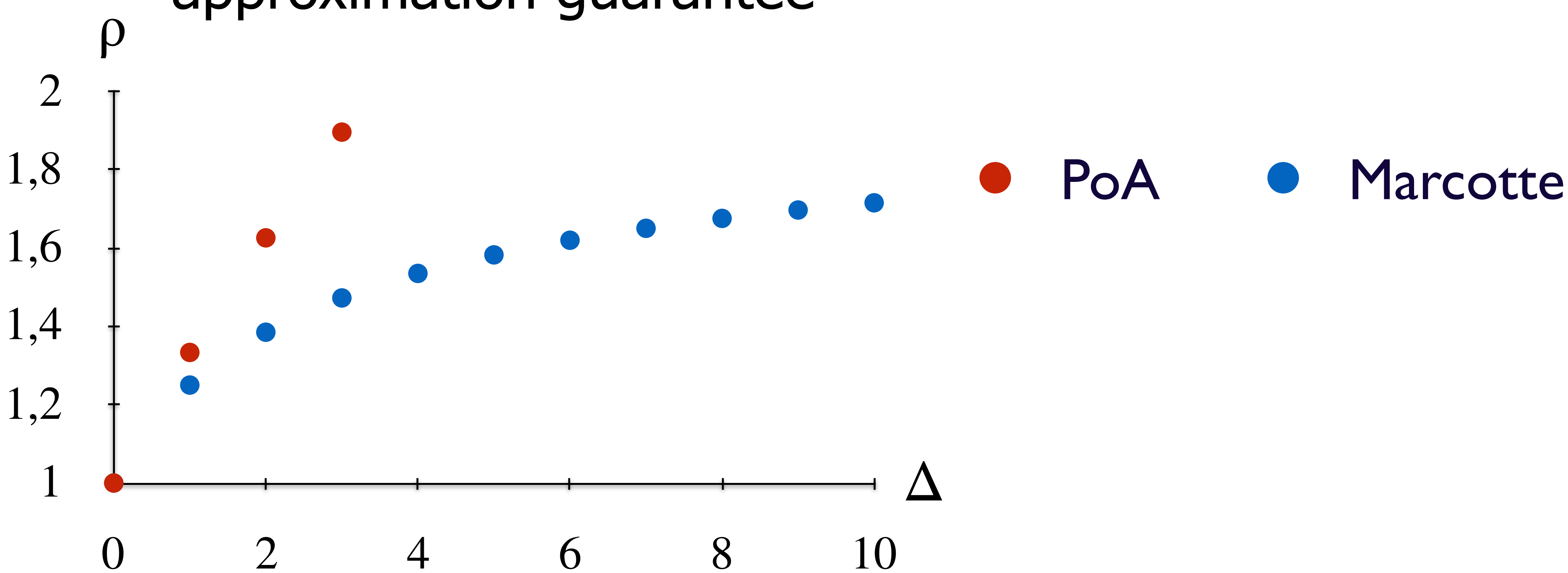
- Solve the relaxation → Compute a Wardrop equilibrium
  - $1/(1-\mu(\mathscr{C}))$-approximation for all $\mathscr{C}$



Known results

- Solve the relaxation → Compute a Wardrop equilibrium
  - $1/(1-\mu(\mathscr{C}))$-approximation for all $\mathscr{C}$
- algorithm of Marcotte for monomials [Marcotte MP '85]
  - $1 + \dfrac{\Delta}{(\Delta+1)^{1+1/\Delta}}$ -approximation for monomials of degree Δ

approximation guarantee



Our contribution

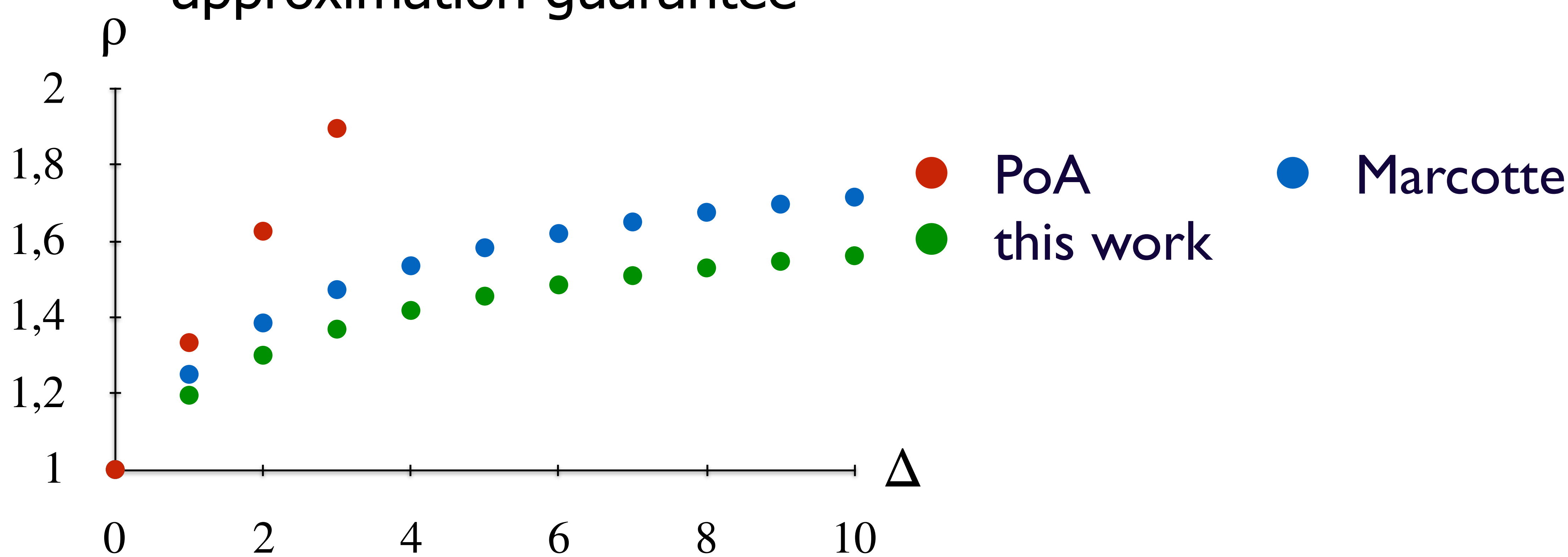
Theorem

[Gairing, Harks, K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

approximation guarantee



Our contribution

Theorem [Gairing, Harks, K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

Our contribution

Theorem [Gairing, Harks, K., ‘14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

Corollary

For affine latencies there is a $49/41$-approximation.

Our contribution

Theorem [Gairing, Harks, K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

Corollary

For affine latencies there is a $49/41$-approximation.

- Proof: $\mu = 1/4$ and $\gamma = 1/2$.

Our contribution

Theorem [Gairing, Harks, K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

Corollary

For affine latencies there is a $49/41$-approximation.

- Proof: $\mu = 1/4$ and $\gamma = 1/2$.

Corollary

For any $\mathscr{C}$ there is a $9/5$-approximation.

Our contribution

Theorem [Gairing, Harks, K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

Corollary

For affine latencies there is a $49/41$-approximation.

- Proof: $\mu = 1/4$ and $\gamma = 1/2$.

Corollary

For any $\mathscr{C}$ there is a $9/5$-approximation.

- Prof: $\mu \leq 1$ and $\gamma \leq 1$ for all $\mathscr{C}$, ρ non-decreasing in μ and γ.

Our algorithm

Input: $G = (N,E)$, (ce)

Output:

- Compute μ,γ
- $(\mathbf{z}, \mathbf{f}) \leftarrow$ Solution of the relaxation
- $$p \leftarrow \frac{\sum_{e\in E} \left(c_e(f_e/z_e)f_e\right)}{\sum_{e\in E} \left(c_e(f_e/z_e)f_e + z_e k_e\right)} \qquad p^* \leftarrow \frac{(\gamma\text{-}\mu+1)^2}{(\gamma\text{-}\mu+1)^2 + 4\mu}$$

Our algorithm

Input: $G = (N,E)$, (ce)

Output:

- Compute μ,γ
- $(\mathbf{z}, \mathbf{f}) \leftarrow$ Solution of the relaxation
- $$p \leftarrow \frac{\sum_{e\in E} \left(c_e(f_e/z_e)f_e\right)}{\sum_{e\in E} \left(c_e(f_e/z_e)f_e + z_e k_e\right)} \qquad p^* \leftarrow \frac{(\gamma-\mu+1)^2}{(\gamma-\mu+1)^2 + 4\mu}$$

$p < p^*$

- let γ_e solve $\left(c_e(x_e)x_e\right)' = c_e(x_e / \gamma_e)$ for $x_e = f_e / z_e$
- $z_e^* \leftarrow \gamma_e z_e \; \forall\, e \in E$
- return $(\mathbf{z}^*, \mathbf{f})$

Our algorithm

Input: $G = (N,E)$, (ce)

Output:

- Compute μ,γ
- $(\mathbf{z}, \mathbf{f}) \leftarrow$ Solution of the relaxation
- $p \leftarrow \dfrac{\sum_{e\in E} \left(c_e(f_e/z_e)f_e\right)}{\sum_{e\in E} \left(c_e(f_e/z_e)f_e + z_e k_e\right)}$ $\qquad p^* \leftarrow \dfrac{(\gamma\text{-}\mu+1)^2}{(\gamma\text{-}\mu+1)^2 + 4\mu}$

$p < p^*$

- let γ_e solve $\left(c_e(x_e)x_e\right)' = c_e(x_e/\gamma_e)$ for $x_e = f_e/z_e$
- $z_e^* \leftarrow \gamma_e z_e \;\forall\, e \in E$
- return $(\mathbf{z}^*, \mathbf{f})$

$p > p^*$

- $\lambda \leftarrow \mu + \sqrt{\mu \frac{p}{1-p}}$
- $\mathbf{f}^* \leftarrow$ Wardrop equilibrium for $\lambda\mathbf{z}^*$
- return $(\lambda\mathbf{z}^*, \mathbf{f}^*)$

Our algorithm

Input: $G = (N,E)$, (ce)

Output:

- Compute μ,γ
- $(\mathbf{z}, \mathbf{f}) \leftarrow$ Solution of the relaxation
- $p \leftarrow \dfrac{\sum_{e\in E} (c_e(f_e/z_e)f_e)}{\sum_{e\in E} (c_e(f_e/z_e)f_e + z_e k_e)}$ $\qquad p^* \leftarrow \dfrac{(\gamma\text{-}\mu+1)^2}{(\gamma\text{-}\mu+1)^2 + 4\mu}$

$p < p^*$

- let γ_e solve $(c_e(x_e)x_e)' = c_e(x_e / \gamma_e)$ for $x_e = f_e / z_e$
- $z_e^* \leftarrow \gamma_e z_e \; \forall\, e \in E$
- return $(\mathbf{z}^*, \mathbf{f})$

[Marcotte, '85]

$p > p^*$

- $\lambda \leftarrow \mu + \sqrt{\mu \frac{p}{1-p}}$
- $\mathbf{f}^* \leftarrow$ Wardrop equilibrium for $\lambda \mathbf{z}^*$
- return $(\lambda \mathbf{z}^*, \mathbf{f}^*)$

Proof idea

Theorem [Gairing, Harks & K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

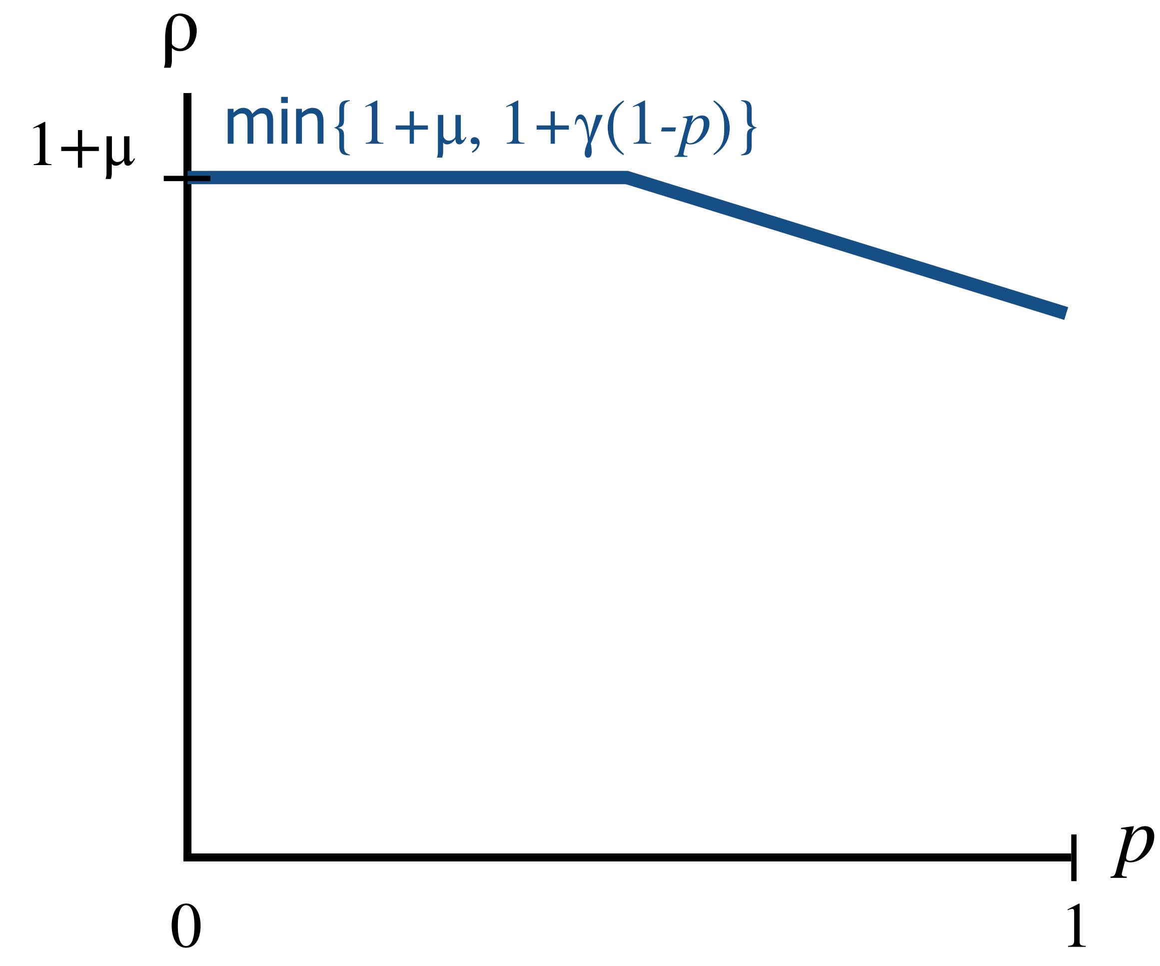
Proof idea

Theorem [Gairing, Harks & K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

- Separate analysis of the two cases:
  - Marcotte's algorithm:
    - $\rho \leq \min\{1+\mu,\ 1+\gamma(1-p)\}$



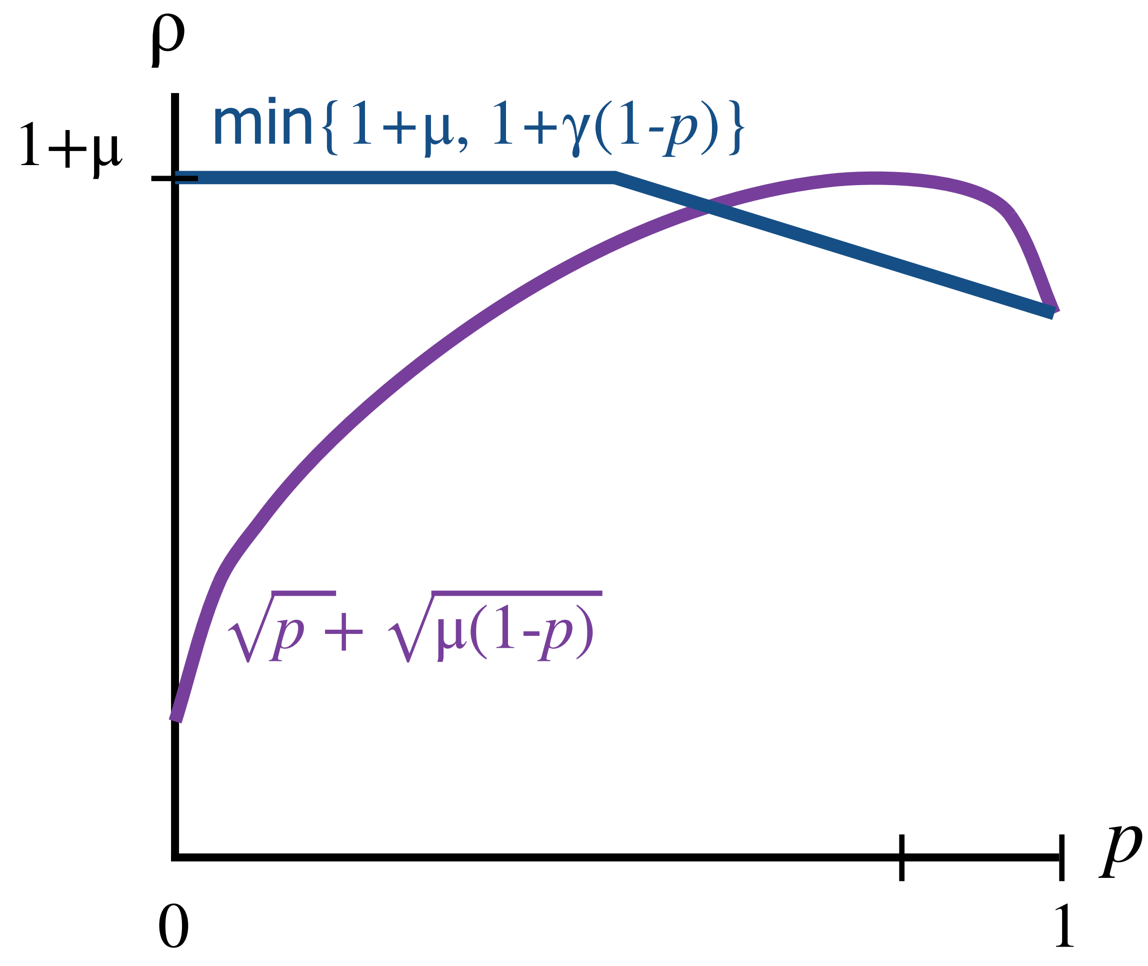
Proof idea

Theorem [Gairing, Harks & K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

- Separate analysis of the two cases:
  - Marcotte's algorithm:
    - $\rho \leq \min\{1+\mu,\ 1+\gamma(1-p)\}$
  - λ-algorithm:
    - $\rho \leq \sqrt{p} + \sqrt{\mu(1-p)}$



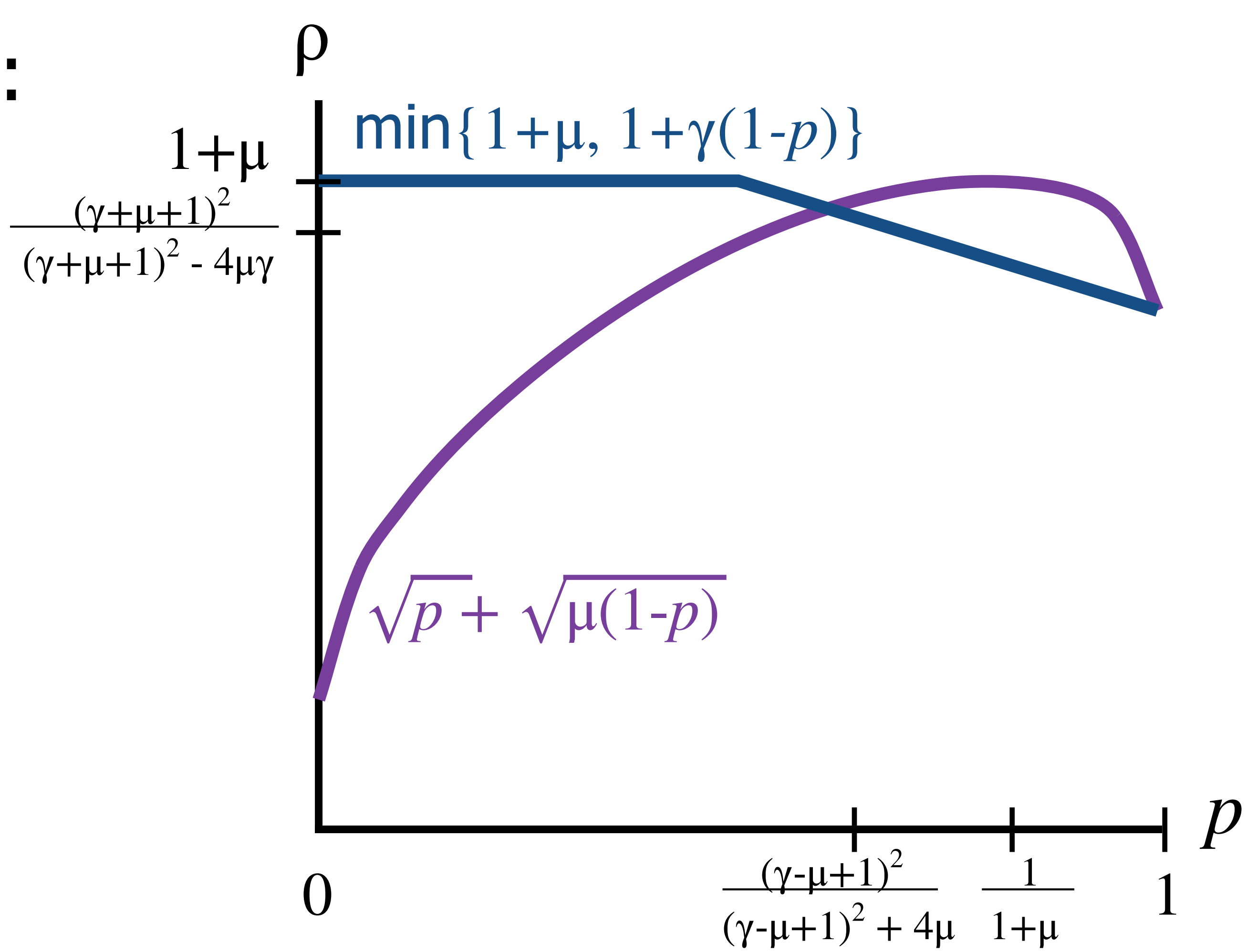
Proof idea

Theorem [Gairing, Harks & K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

- Separate analysis of the two cases:
  - Marcotte's algorithm:
    - $\rho \leq \min\{1+\mu, 1+\gamma(1-p)\}$
  - λ-algorithm:
    - $\rho \leq \sqrt{p} + \sqrt{\mu(1-p)}$



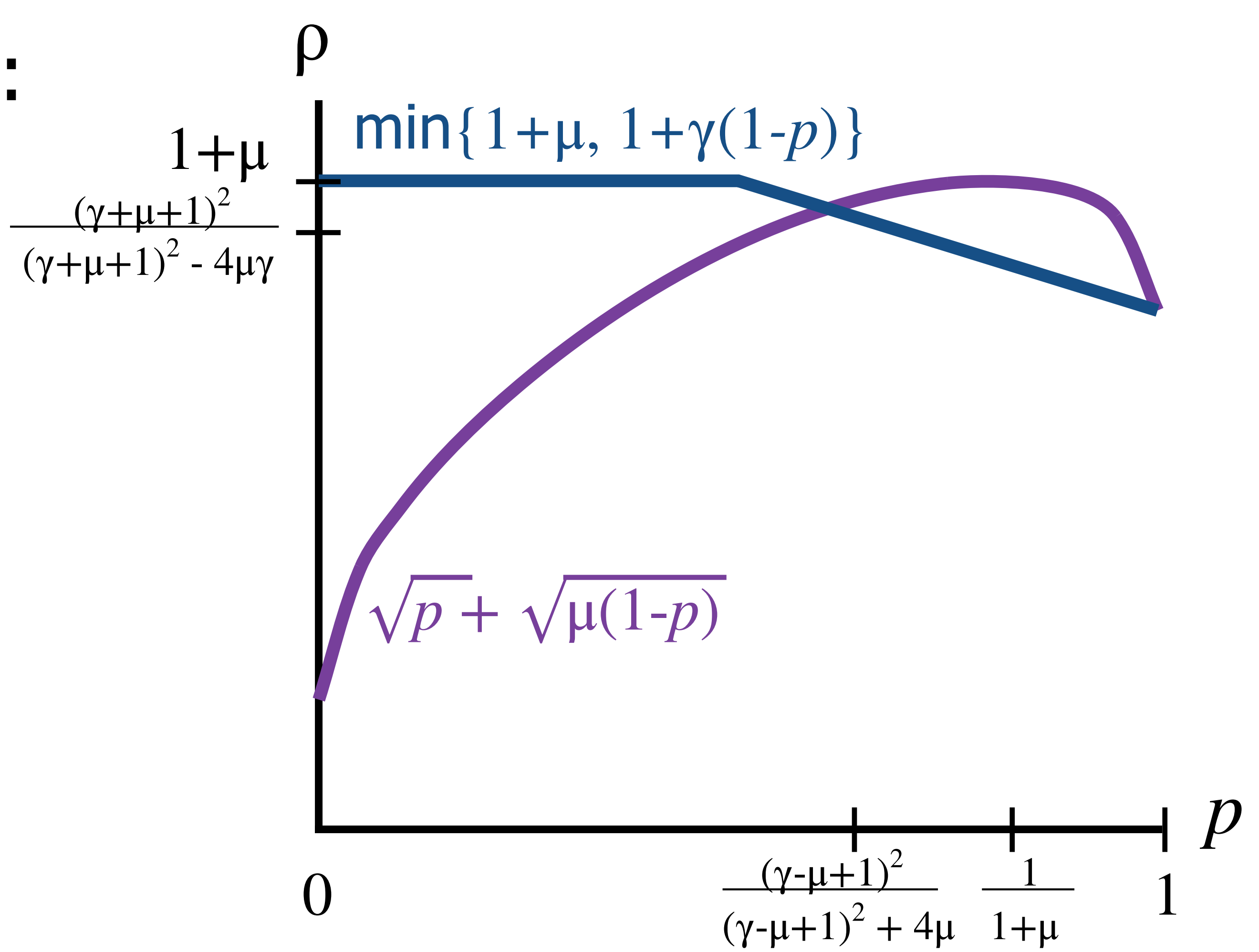
Proof idea

Theorem [Gairing, Harks & K., '14]

There is a ρ-approximation with

$$\rho \leq \frac{(\gamma+\mu+1)^2}{(\gamma+\mu+1)^2 - 4\mu\gamma}.$$

- Separate analysis of the two cases:
  - Marcotte's algorithm:
    - $\rho \leq \min\{1+\mu, 1+\gamma(1-p)\}$
  - λ-algorithm:
    - $\rho \leq \sqrt{p} + \sqrt{\mu(1-p)}$



Approximation guarantee

Lemma

Approximation factor of Marcotte's Algorithm $\leq 1+\mu$.

- Let $x_e = f_e^*/z_e^*$
- by local optimality of z_e^*, we have $k_e = c_e'(x_e)x_e^2$

$$C(\mathbf{f}^*, \mathbf{z}^*) = \sum_{e \in E} c_e(f_e^*/z_e^*)f_e^* + k_e z_e^* = \sum_{e \in E} \Big(c_e(x_e) + c_e'(x_e)x_e\Big) f_e^*$$

- $\mu = \max_{\gamma} \Big(\gamma\big(1 - \frac{c_e(\gamma x)}{c_e(x)}\big)\Big) = \gamma \frac{c_e'(x)x}{c_e(x) + c_e'(x)x}$ s.t. $c_e(x/\gamma) = c_e(x) + c_e'(x)x$

$$\begin{aligned} C(\mathbf{f}^*, \mathbf{z}) &= \sum_{e \in E} c_e(x_e/\gamma_e)f_e^* + k_e \gamma_e z_e^* \\ &= \sum_{e \in E} \Big(c_e(x_e) + c_e'(x_e)x_e\Big) f_e^* + \gamma_e c_e'(x_e)x_e f_e^* \\ &\leq (1+\mu)C(\mathbf{f}^*, \mathbf{z}^*) \end{aligned}$$

Conclusion

- Many networks used by selfish users
- Network design with equilibrium constraints harder than regular network design
- Approximation algorithms based on relaxation
  - 49/41-approximation for affine costs
  - 8/5-approximation for general costs

- Thank you