Spins, Games and Networks

Congestion and traffic routing

Sushmita Gupta

23rd December, 2024

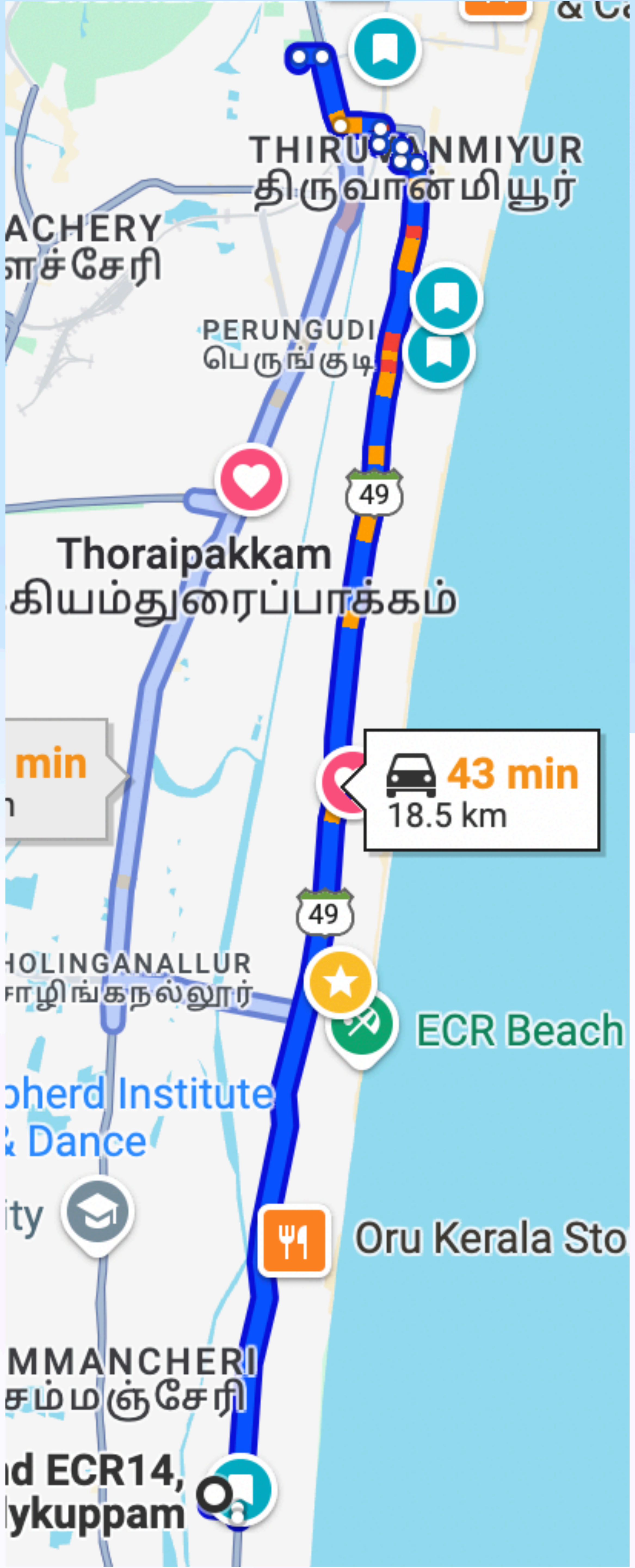
Games in the wild

Games in the wild

So what is a (congestion) game ?

So what is a (congestion) game ?

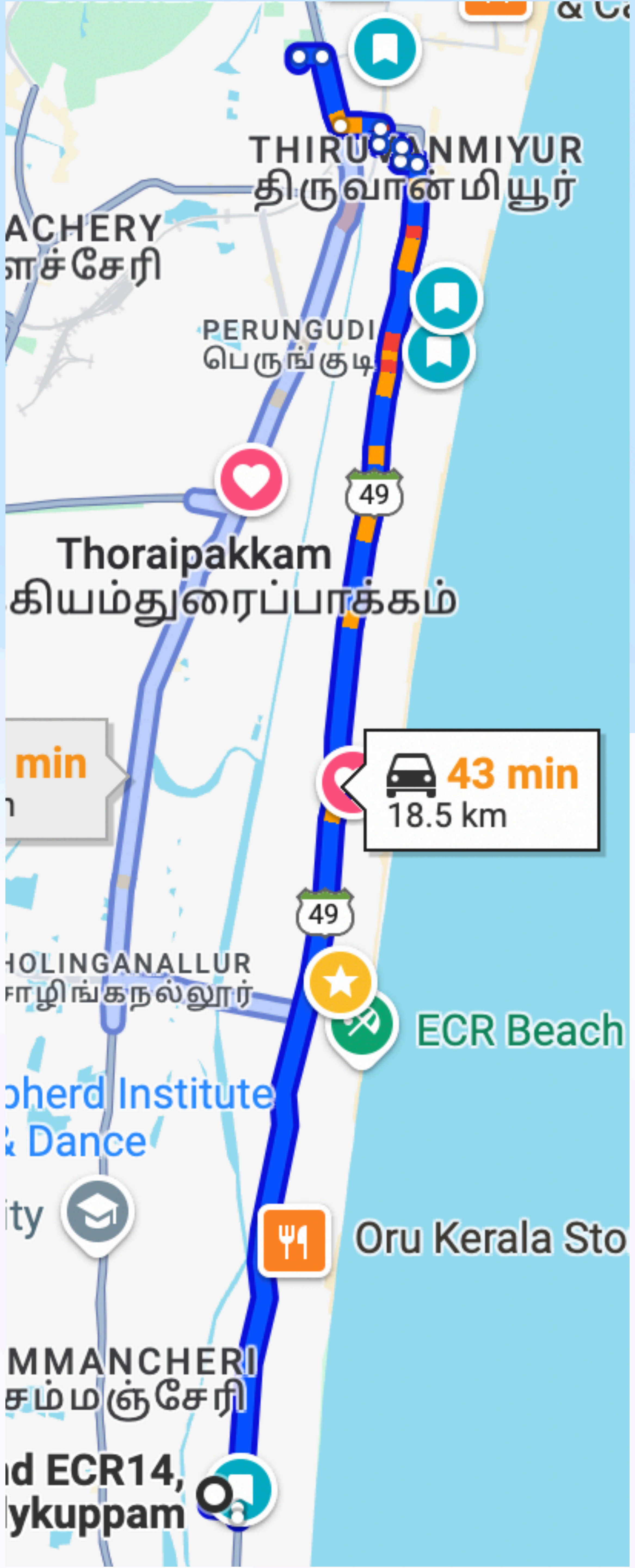
This map will give us the answer



So what is a (congestion) game ?

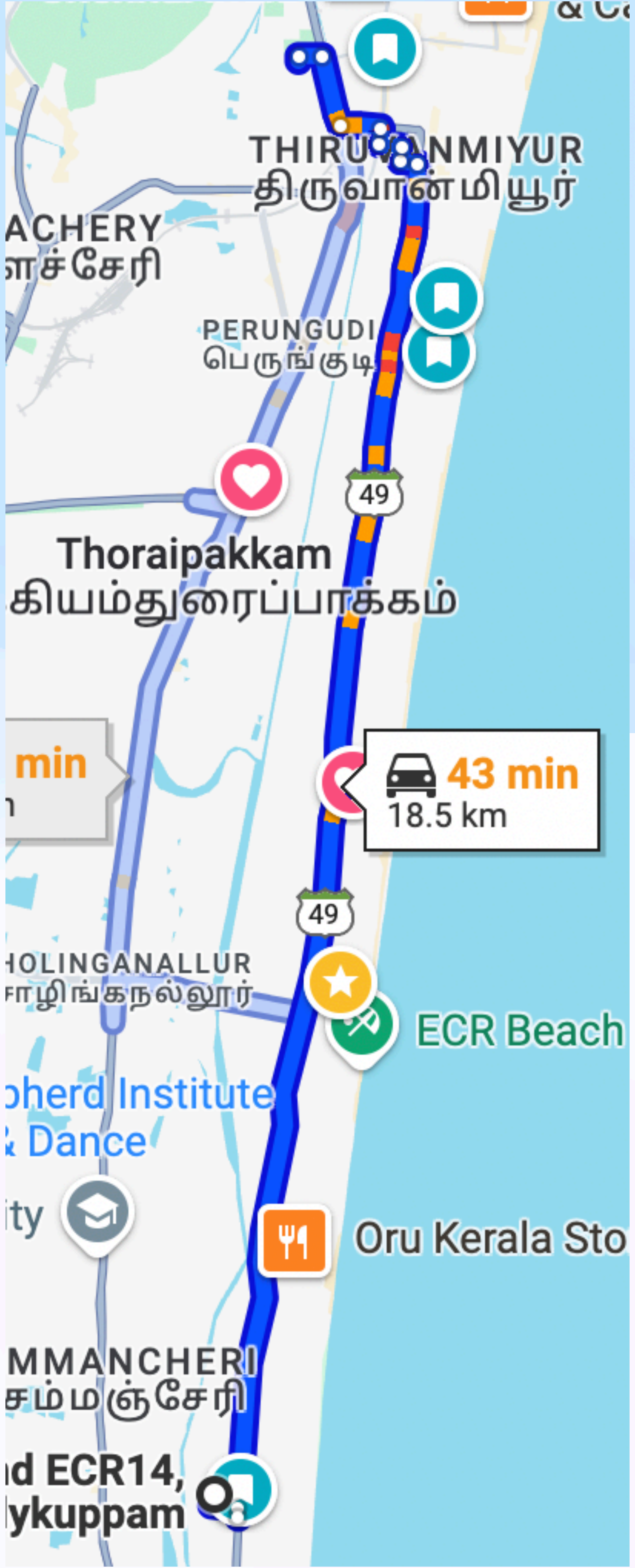
This map will give us the answer

When a situation consists of <u>entities</u>:



So what is a (congestion) game ?

This map will give us the answer



When a situation consists of <u>entities</u>:

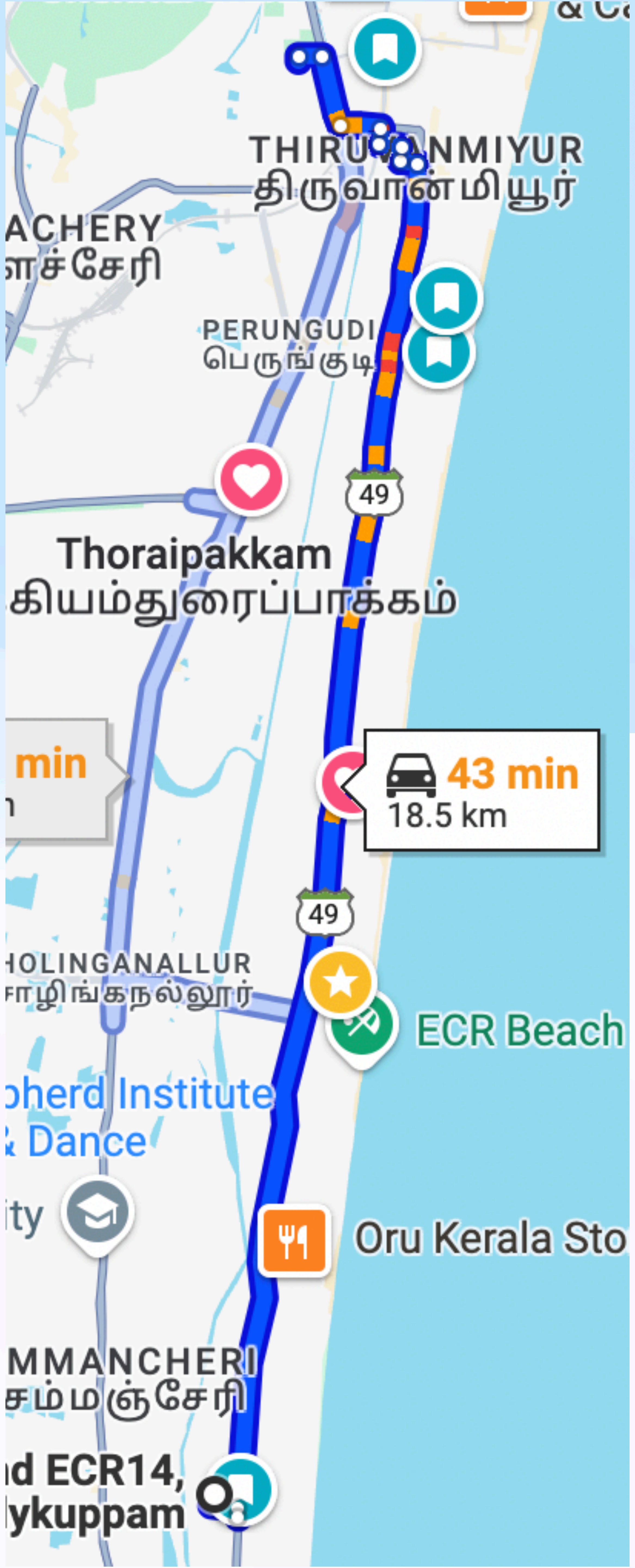
- whose behaviour is not entirely predictable

So what is a (congestion) game ?

This map will give us the answer

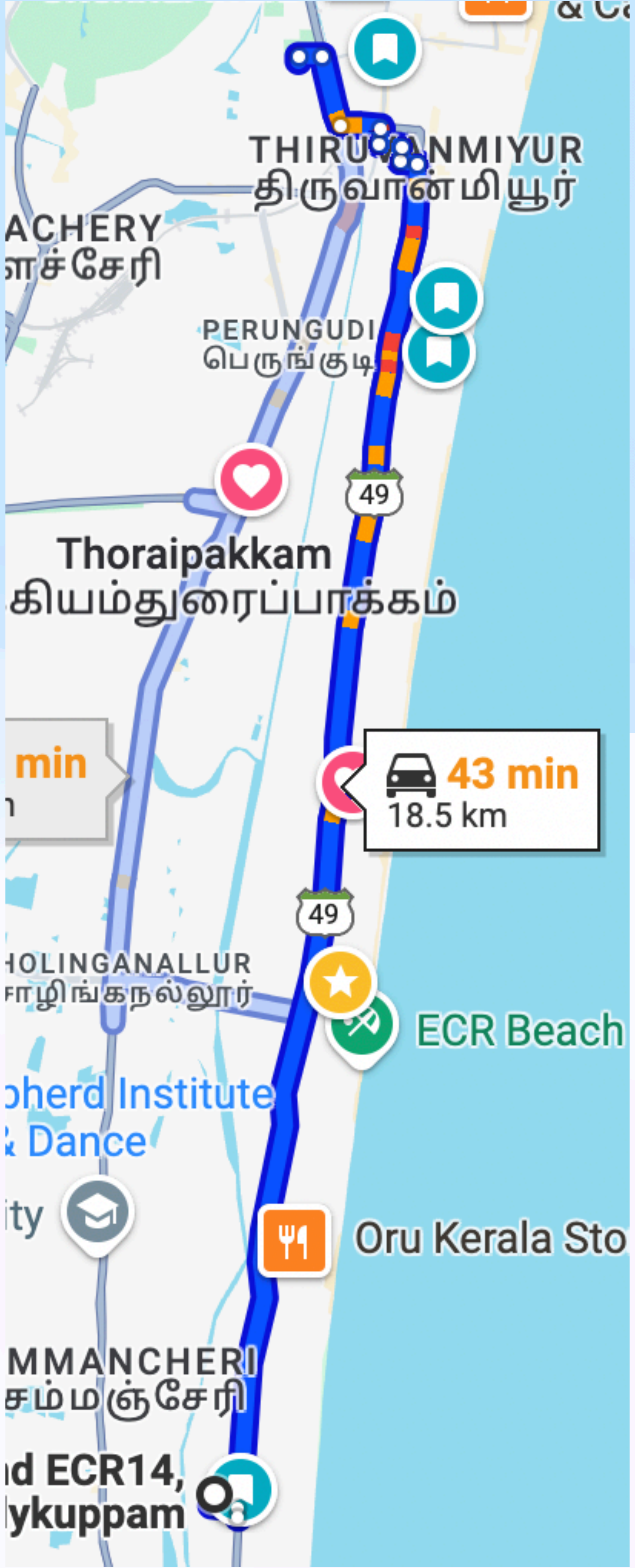
When a situation consists of <u>entities</u>:

- whose behaviour is not entirely predictable
- whose outcome depends on the behaviour of the other entires



So what is a (congestion) game ?

This map will give us the answer



When a situation consists of entities:

- whose behaviour is not entirely predictable
- whose outcome depends on the behaviour of the other entires
- And its behaviour can differ based on (knowledge) of other people's behavior

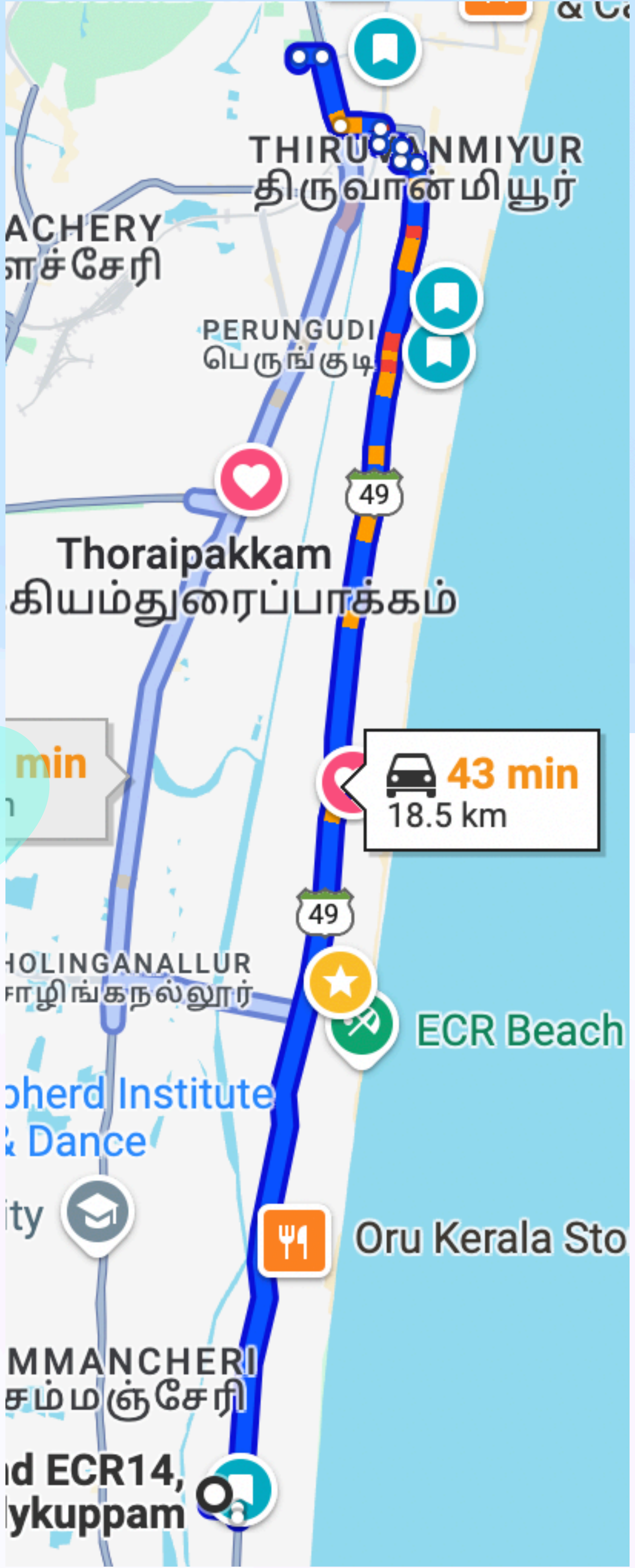
So what is a (congestion) game ?

This map will give us the answer

When a situation consists of <u>entities</u>:

- whose behaviour is not entirely predictable
- whose outcome depends on the behaviour of the other entires
- And its behaviour can differ based on (knowledge) of other people's behavior

Strategic Behavior



So what is a (congestion) game ?

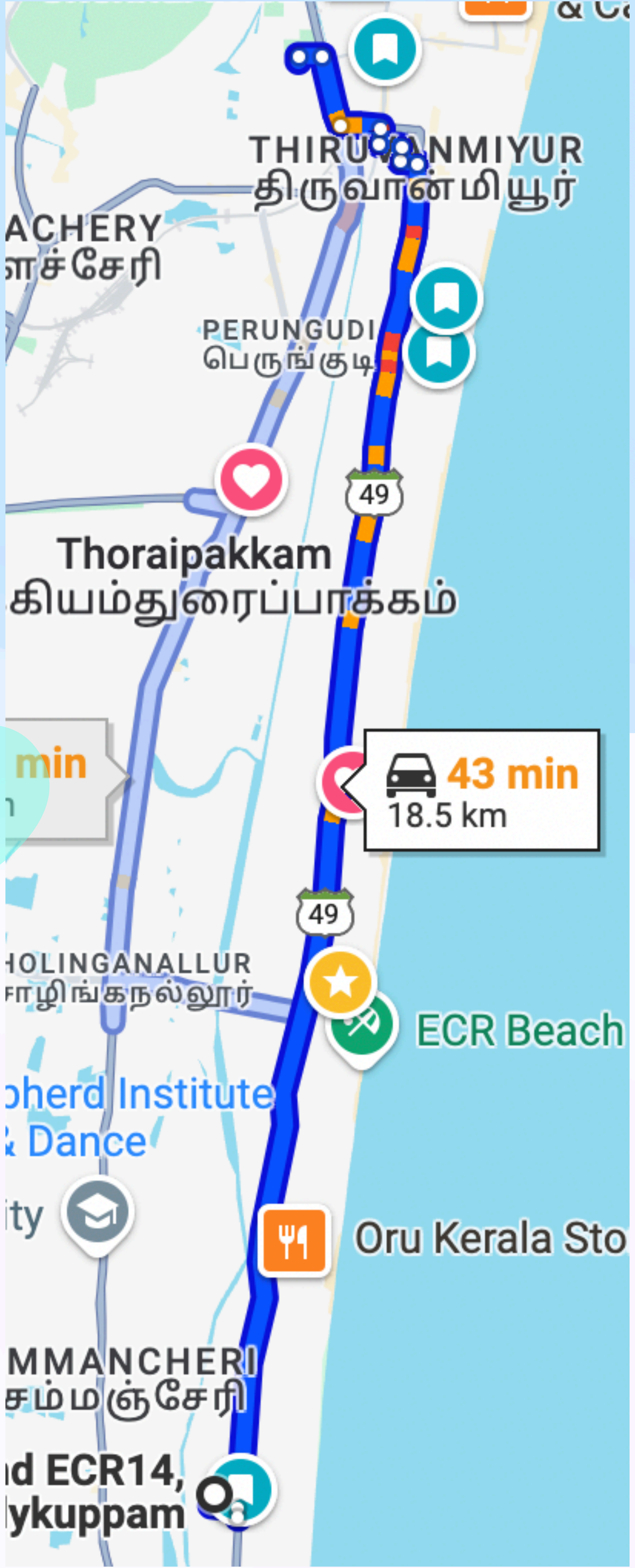
This map will give us the answer

When a situation consists of <u>entities</u>:

- whose behaviour is not entirely predictable
- whose outcome depends on the behaviour of the other entires
- And its behaviour can differ based on (knowledge) of other people's behavior

Strategic Behavior

When the outcome of one depends on the actions of other...we have a game!



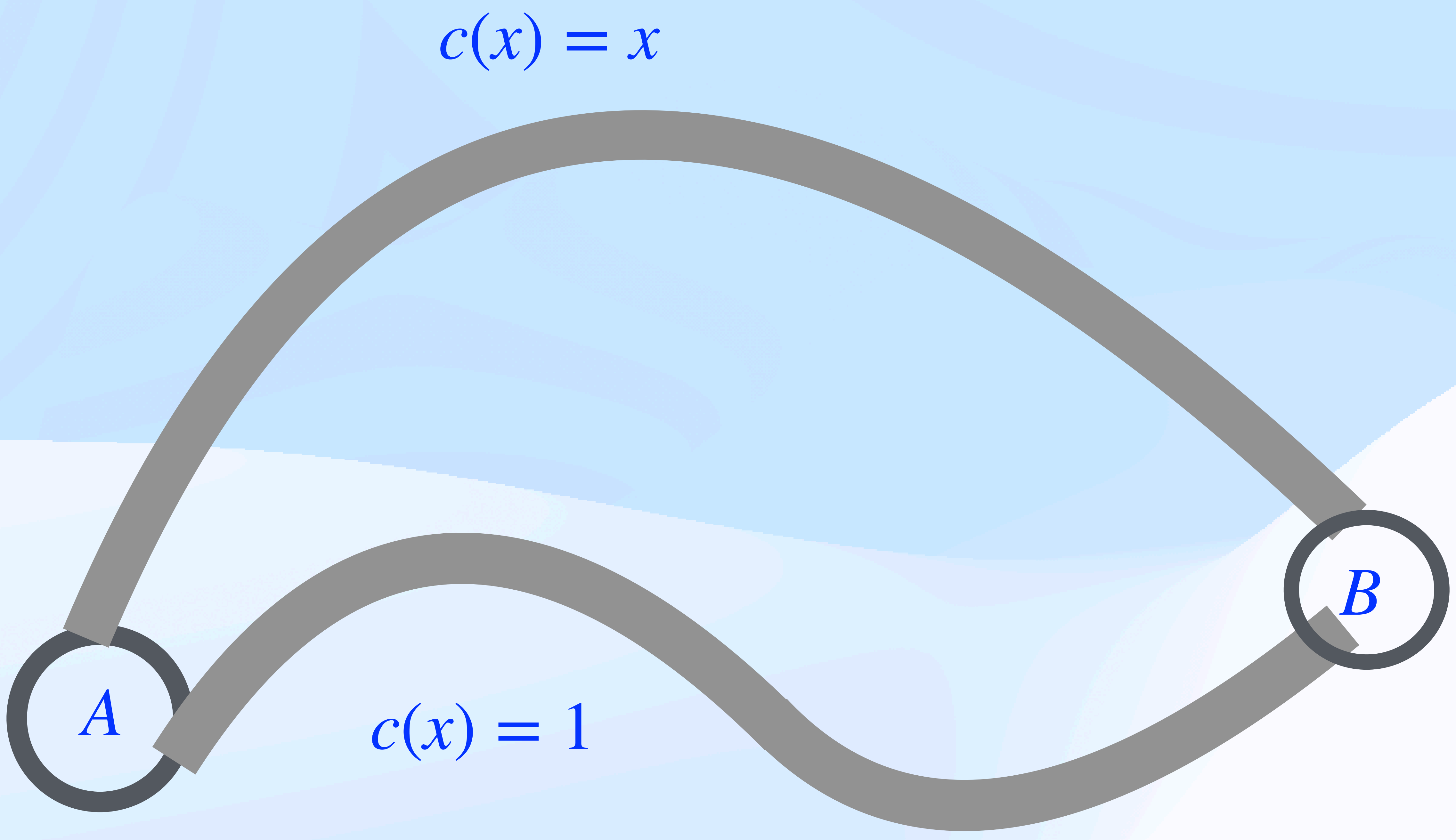
Let's try to understand this phenomenon

Let's try to understand this phenomenon

Pigou's Network 1920

Let's try to understand this phenomenon

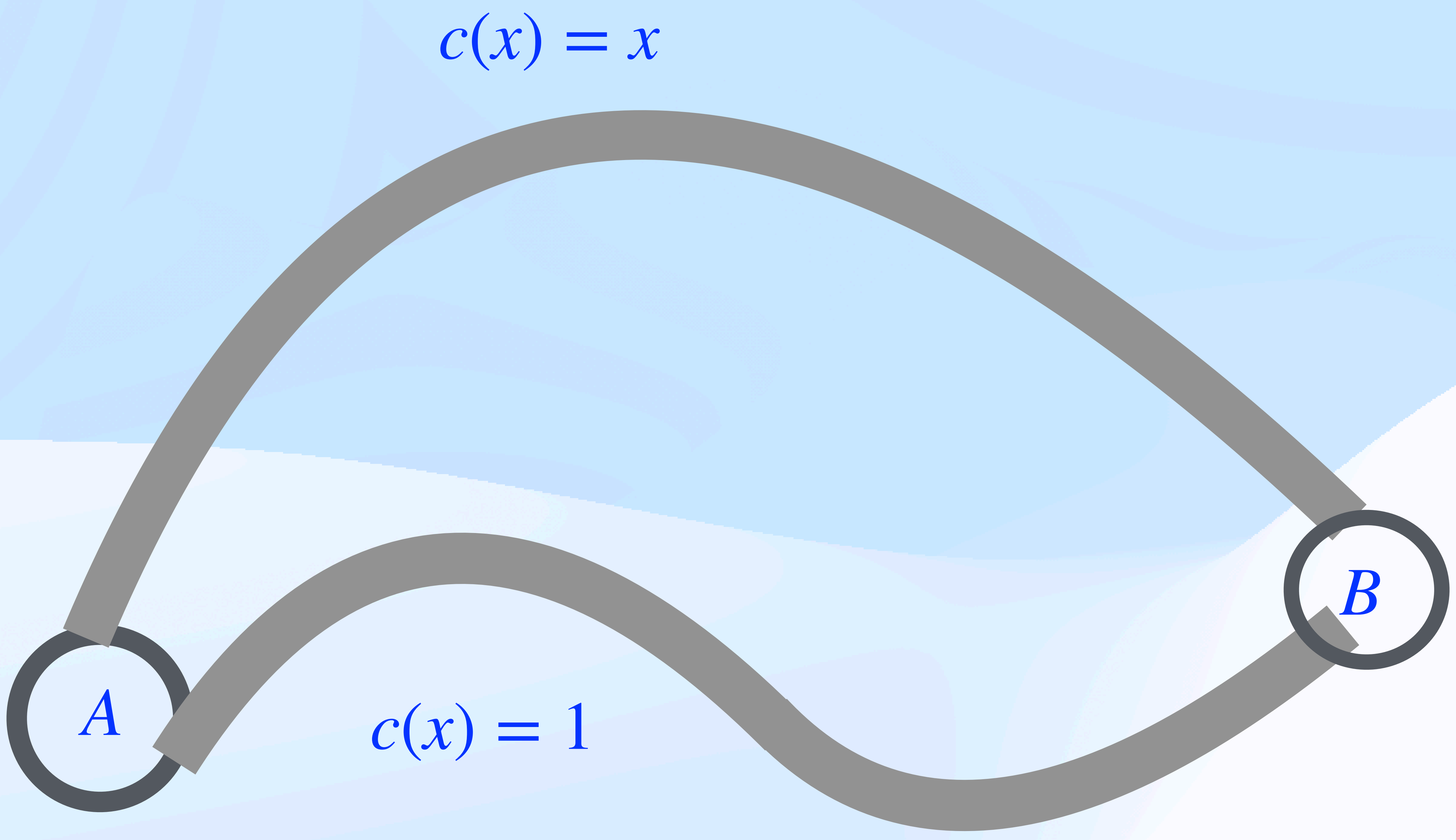
Pigou's Network 1920



Let's try to understand this phenomenon

Pigou's Network 1920

1 unit of traffic

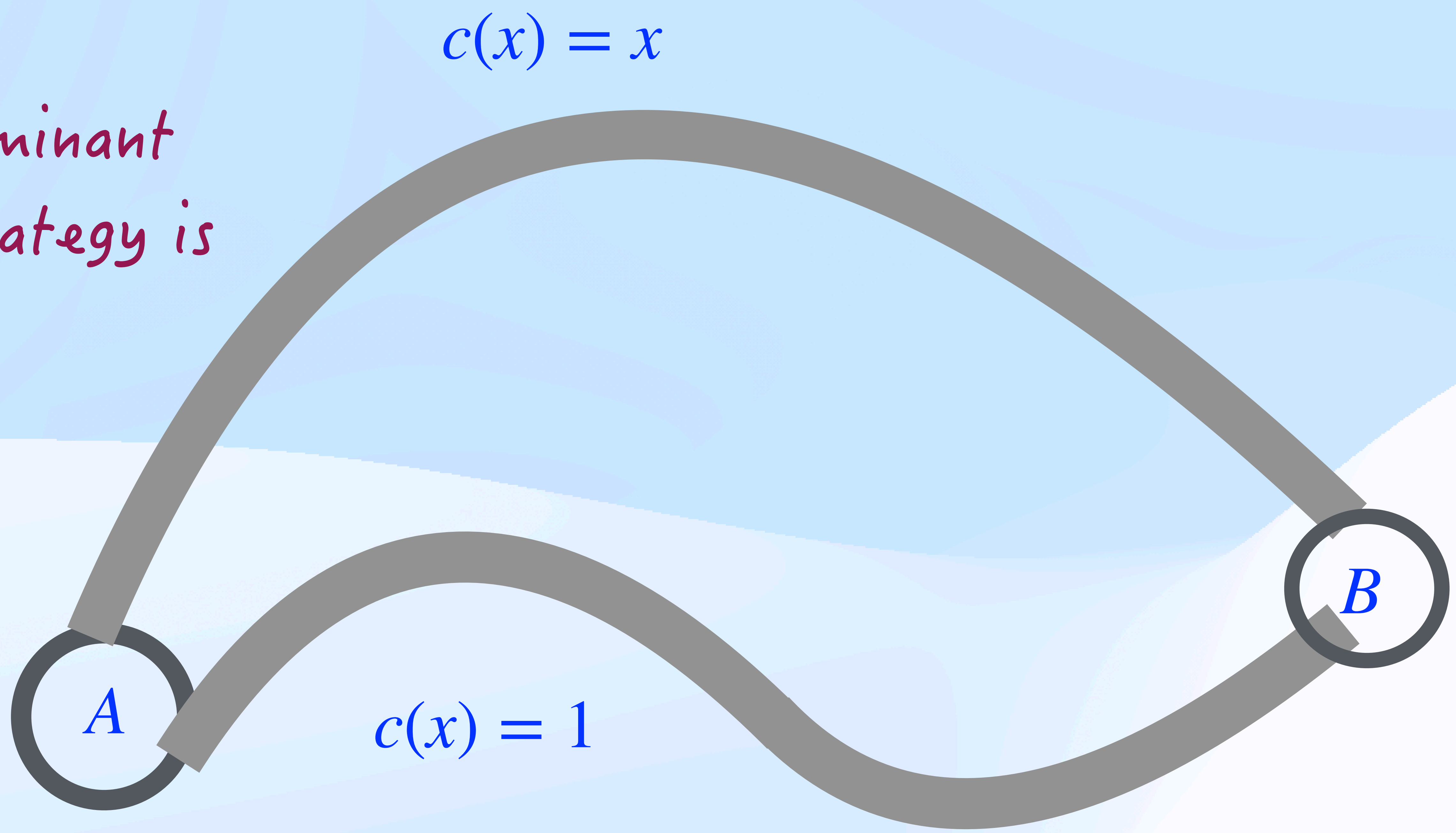


Pigou's Network 1920

Let's try to understand this phenomenon

1 unit of traffic

Dominant Strategy is

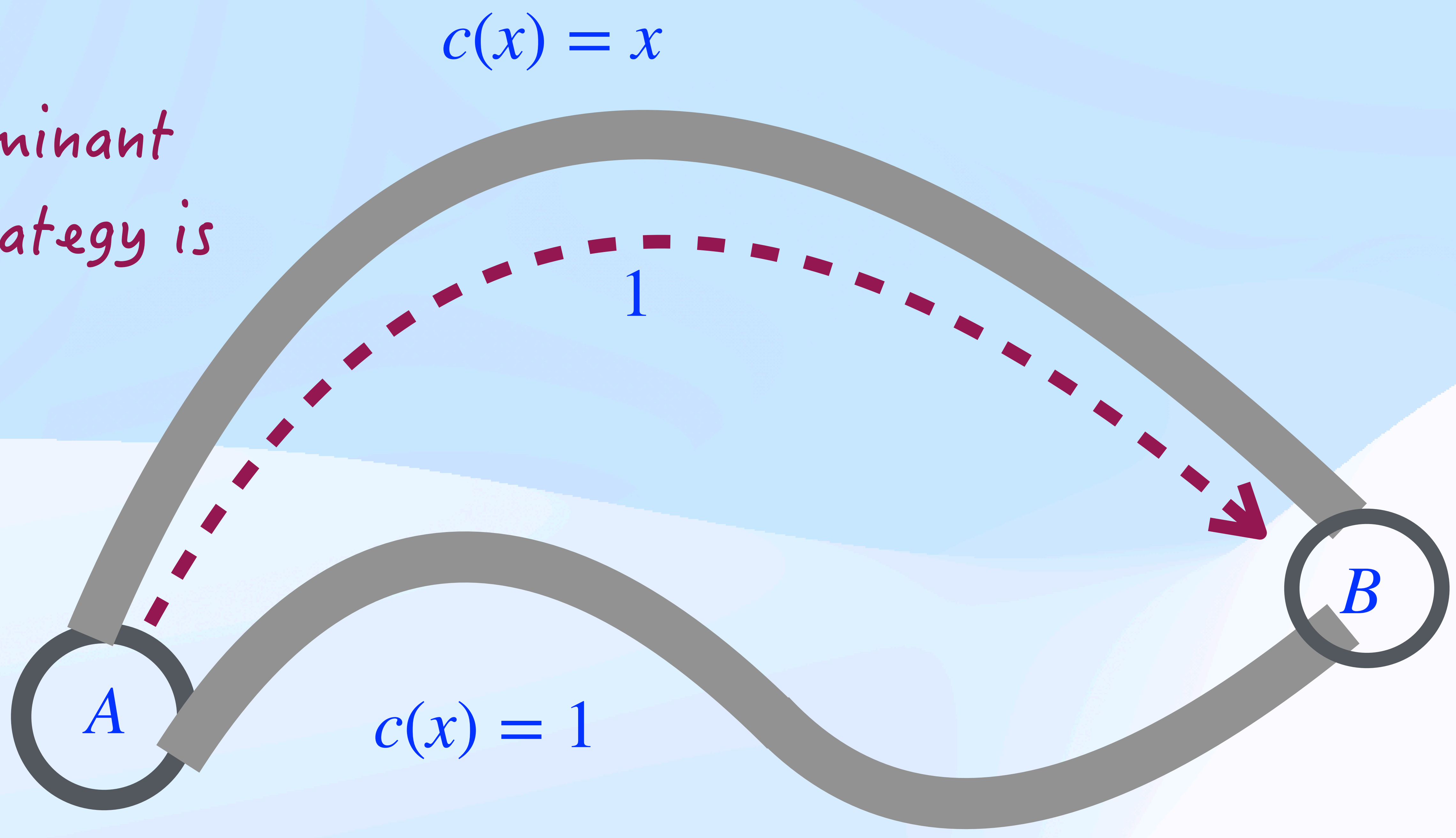


Pigou's Network 1920

Let's try to understand this phenomenon

1 unit of traffic

Dominant Strategy is

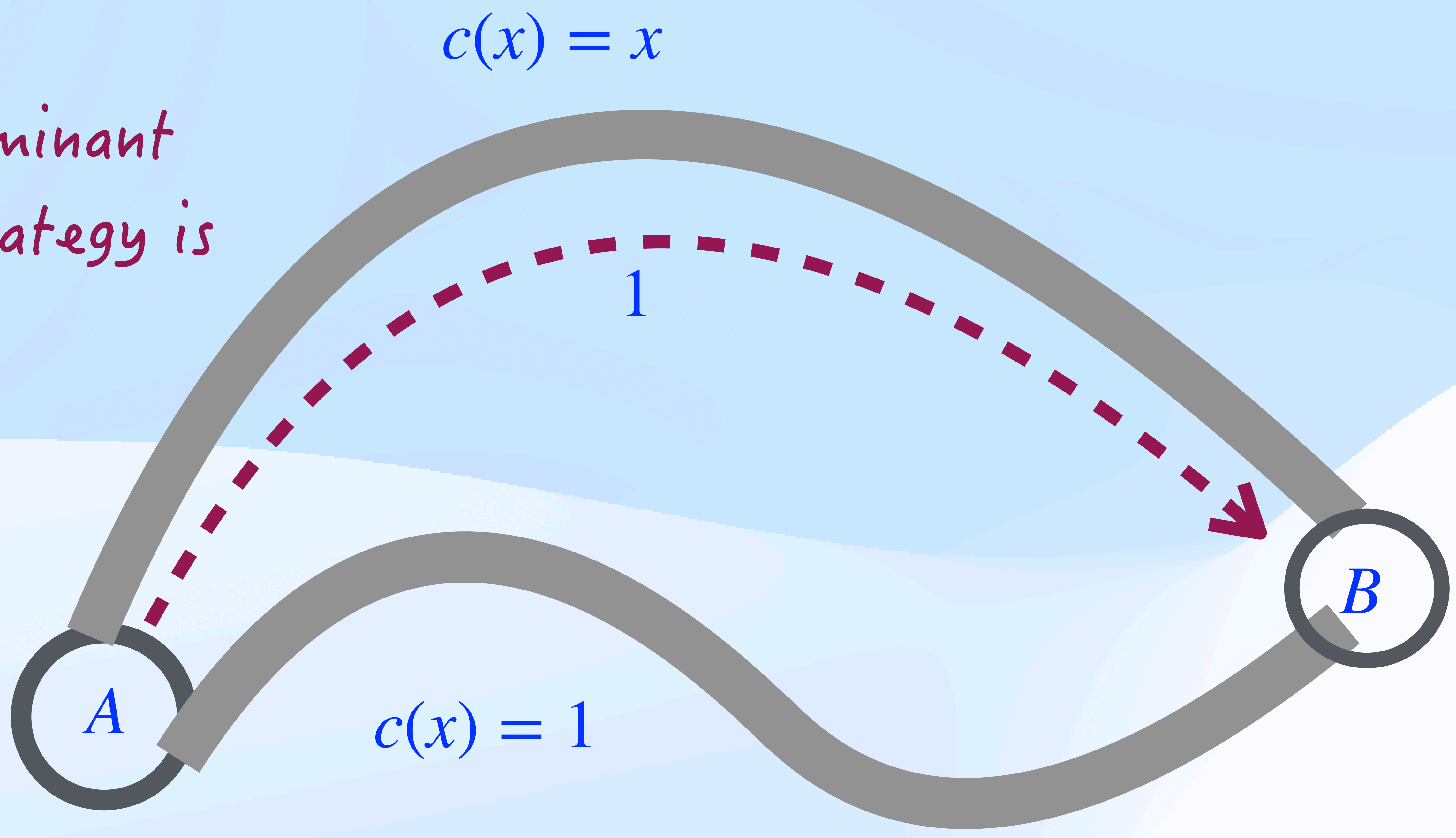


Let's try to understand this phenomenon

Pigou's Network 1920

1 unit of traffic

Dominant Strategy is



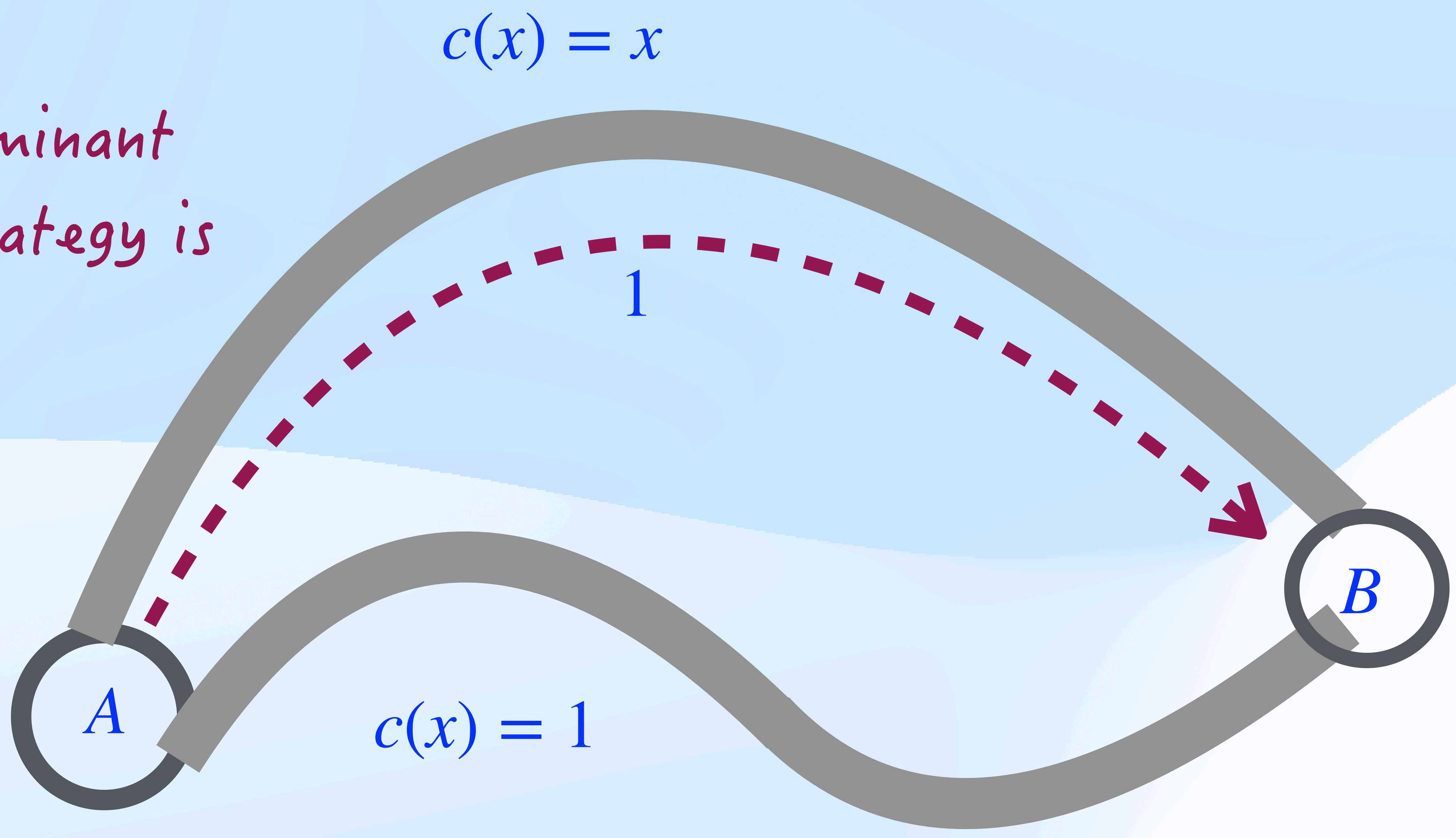
Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

Let's try to understand this phenomenon

Pigou's Network 1920

1 unit of traffic

Dominant Strategy is



Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

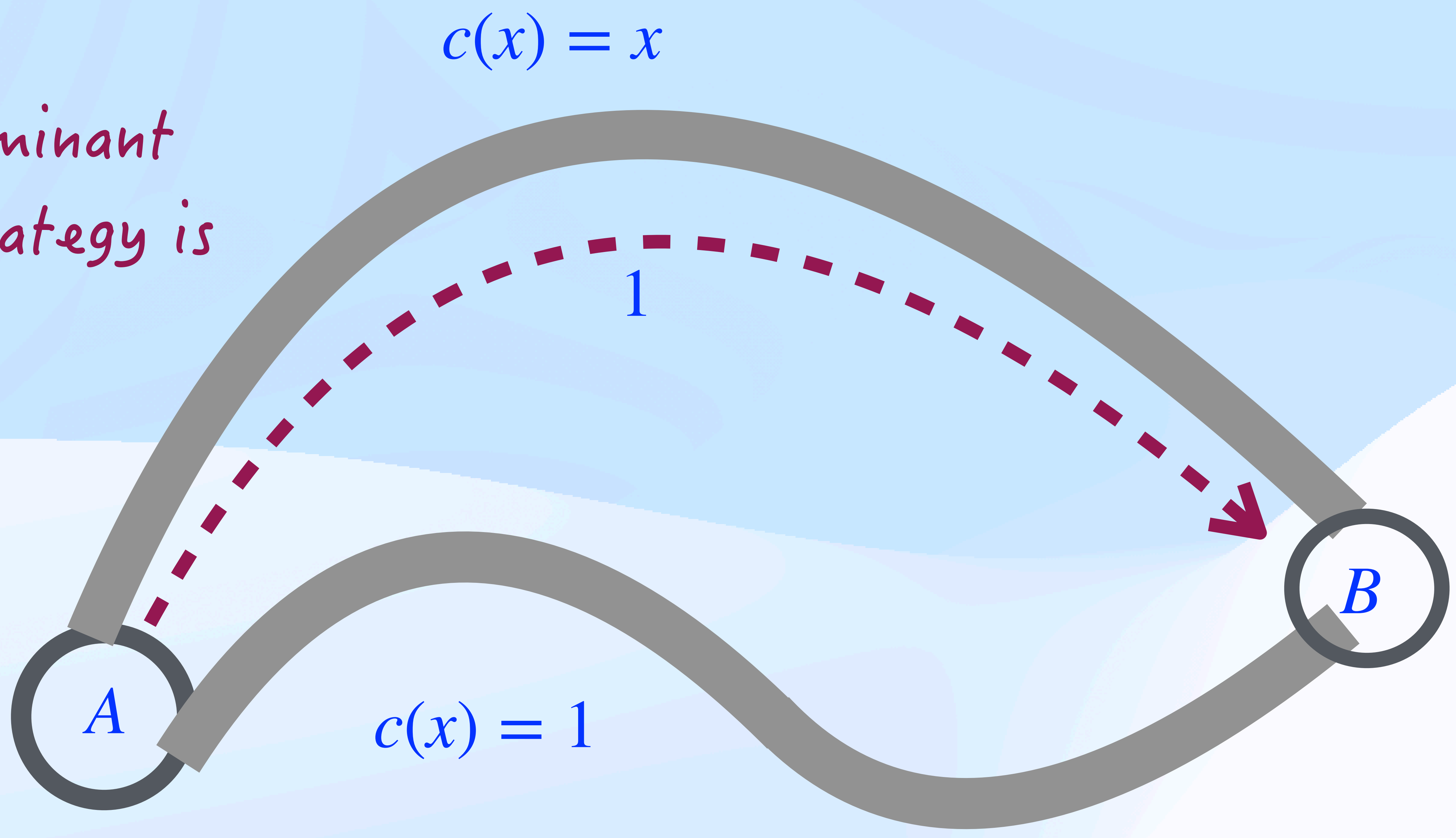
Q: Is this the ideal outcome ?

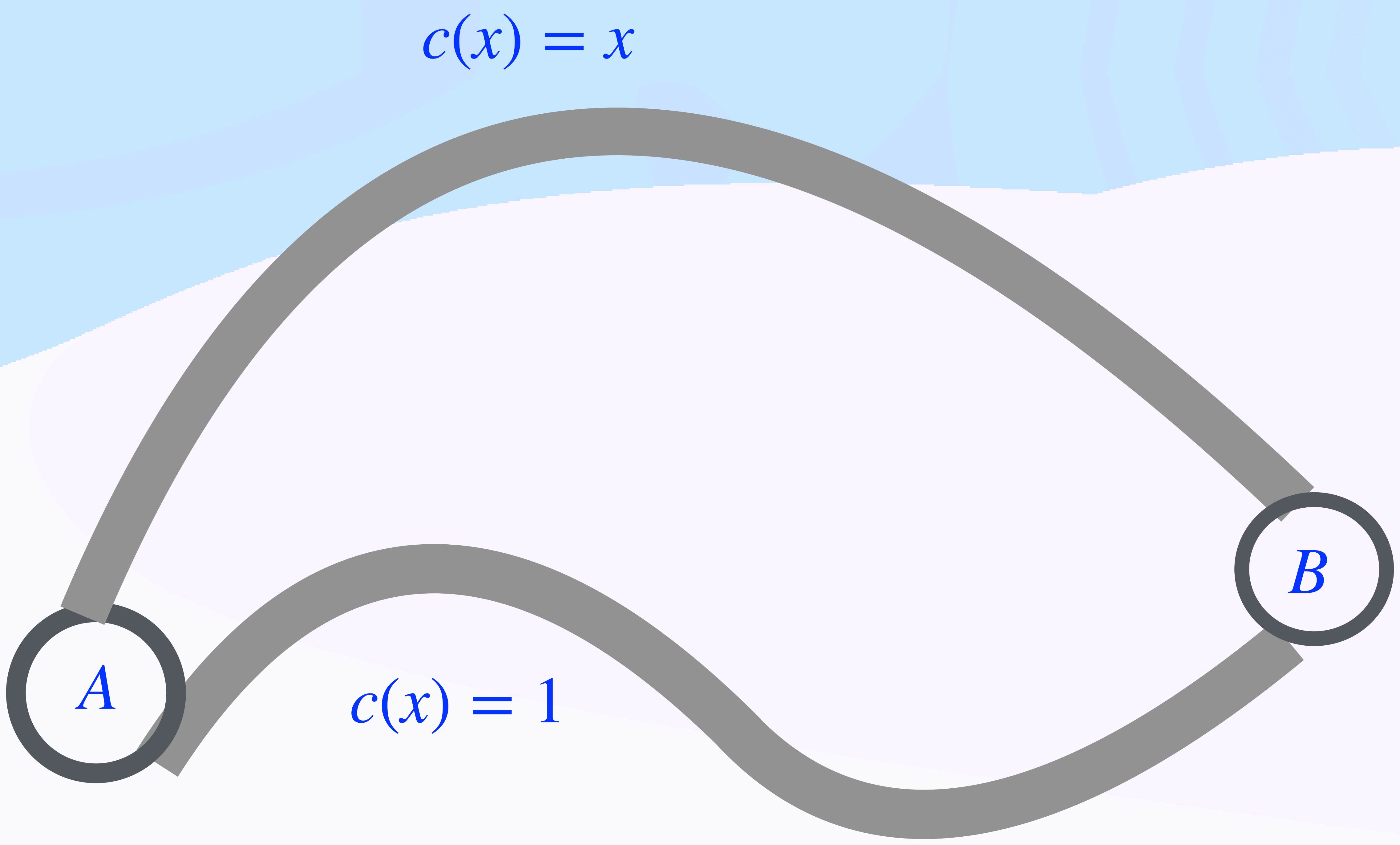
Pigou's Network 1920

Let's try to understand this phenomenon

1 unit of traffic

Dominant Strategy is





Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

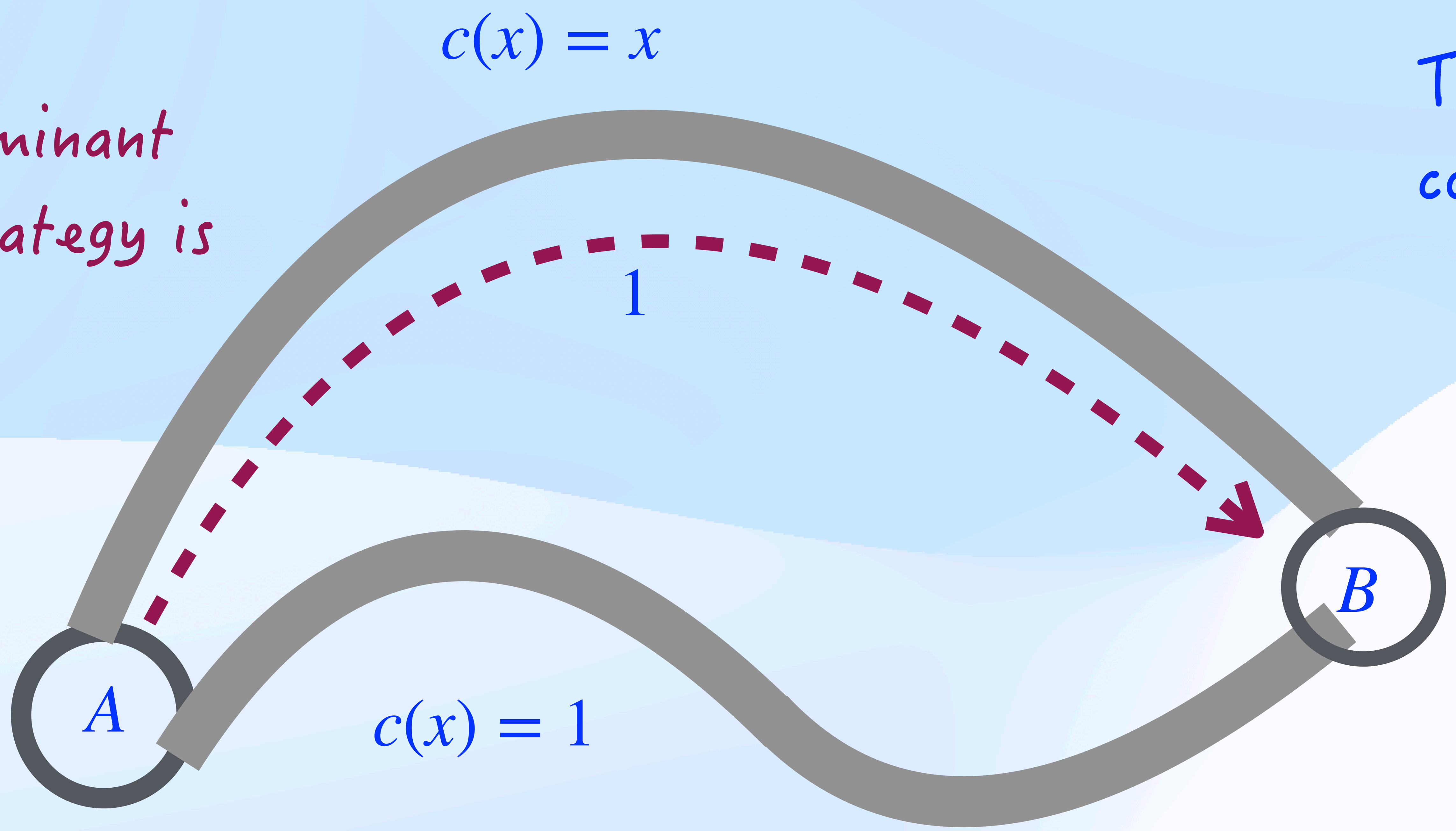
Q: Is this the ideal outcome ?

Let's try to understand this phenomenon

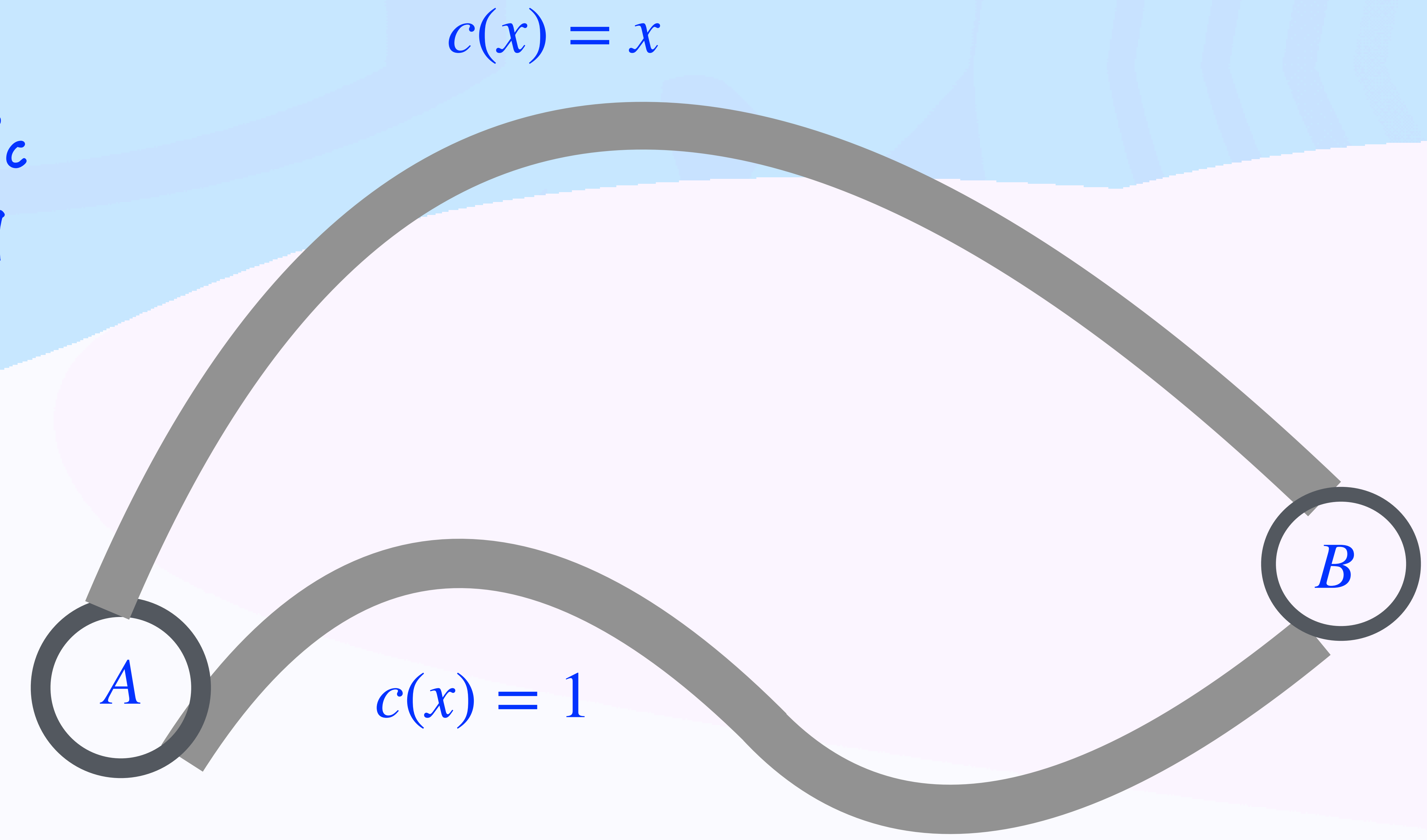
Pigou's Network 1920

1 unit of traffic

Dominant Strategy is



Traffic control



Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

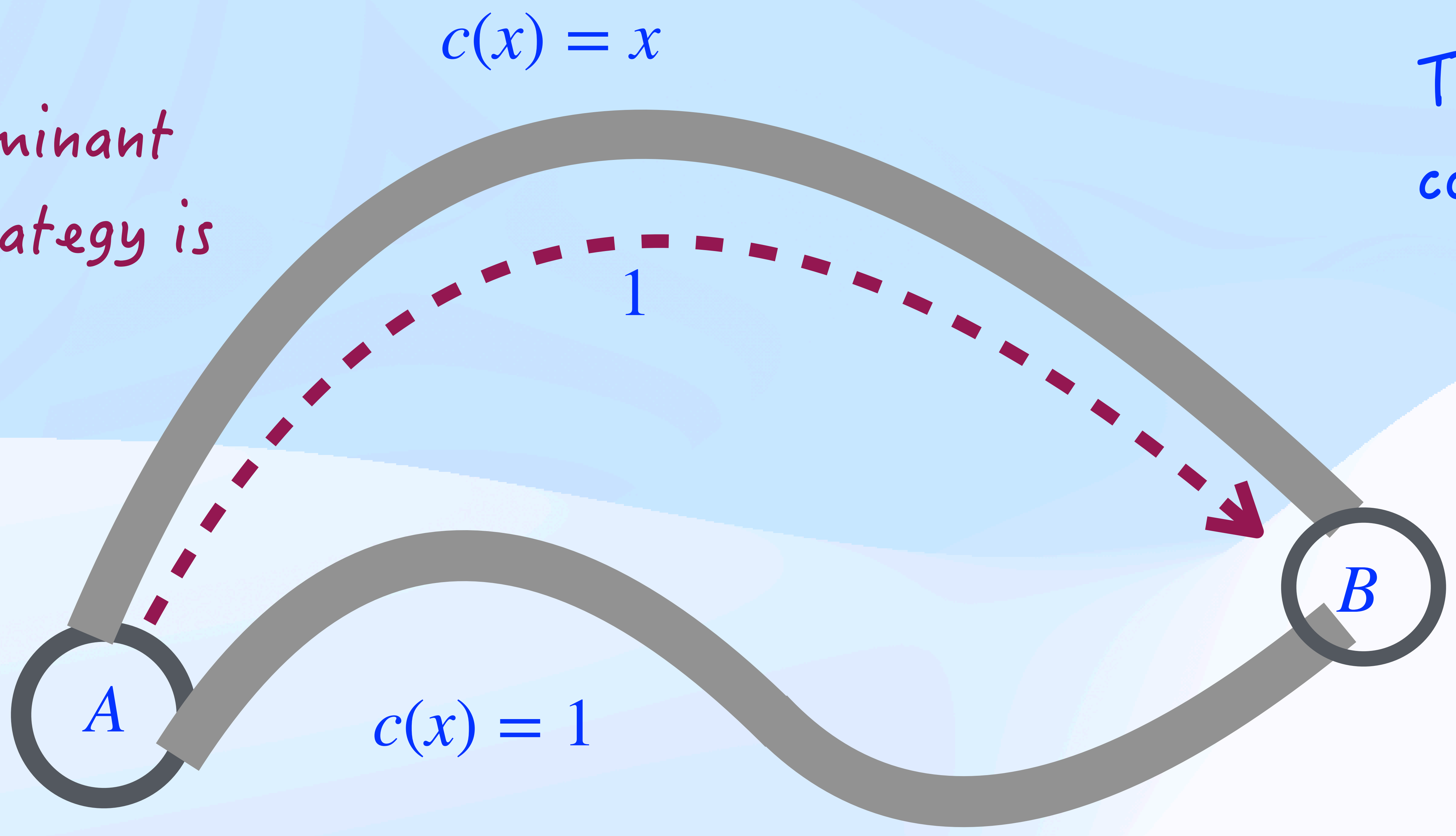
Q: Is this the ideal outcome ?

Pigou's Network 1920

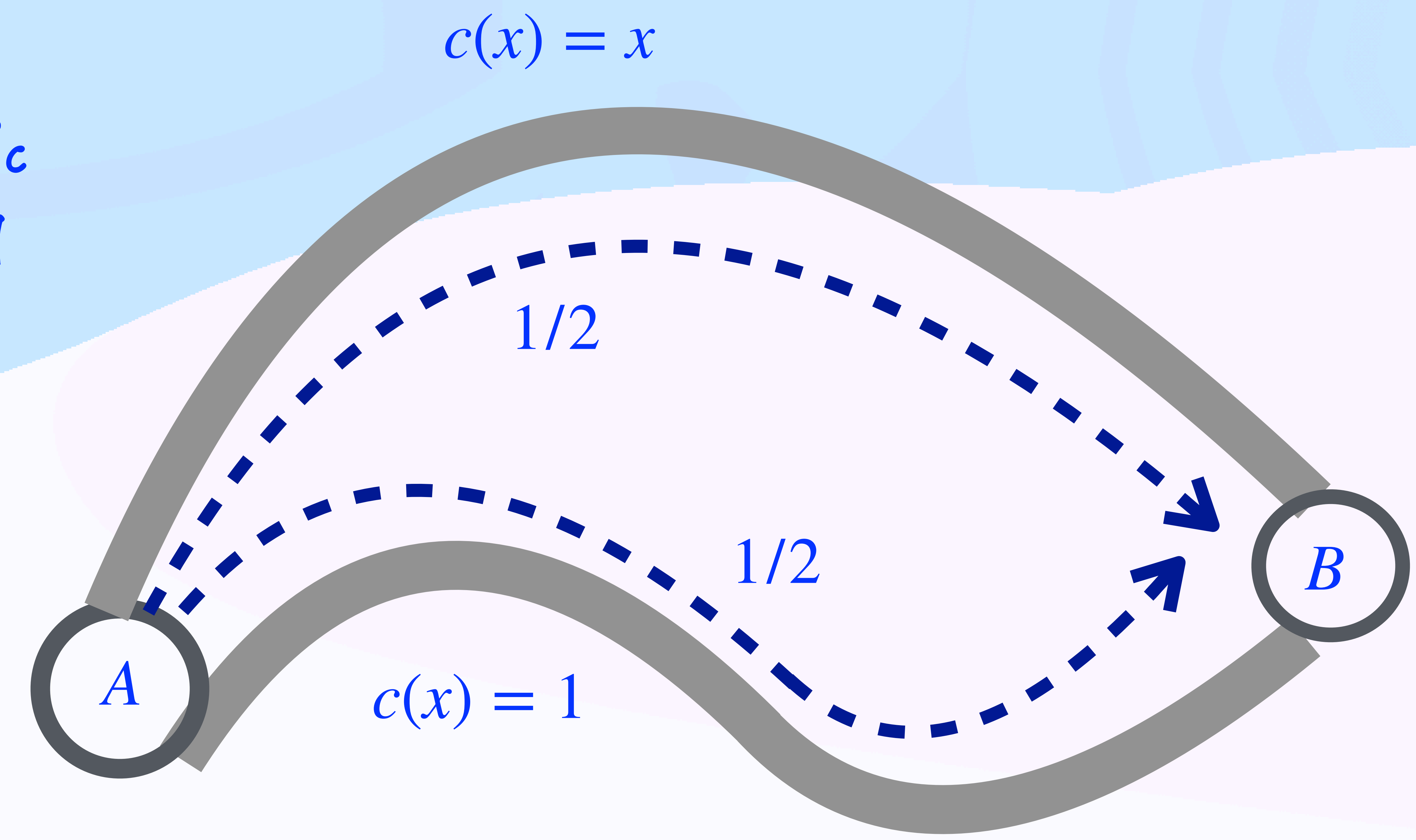
Let's try to understand this phenomenon

1 unit of traffic

Dominant Strategy is



Traffic control



Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

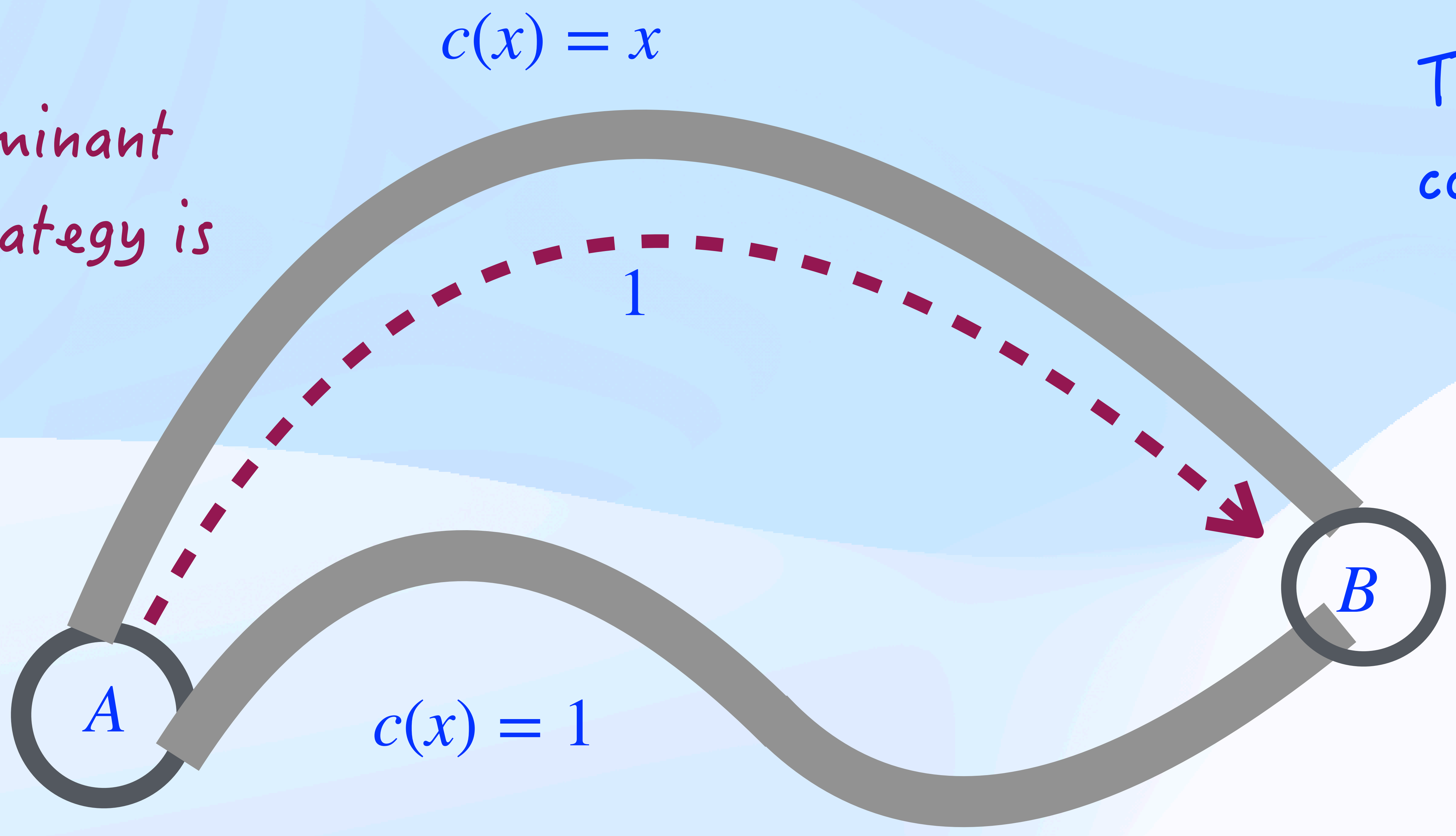
Q: Is this the ideal outcome ?

Let's try to understand this phenomenon

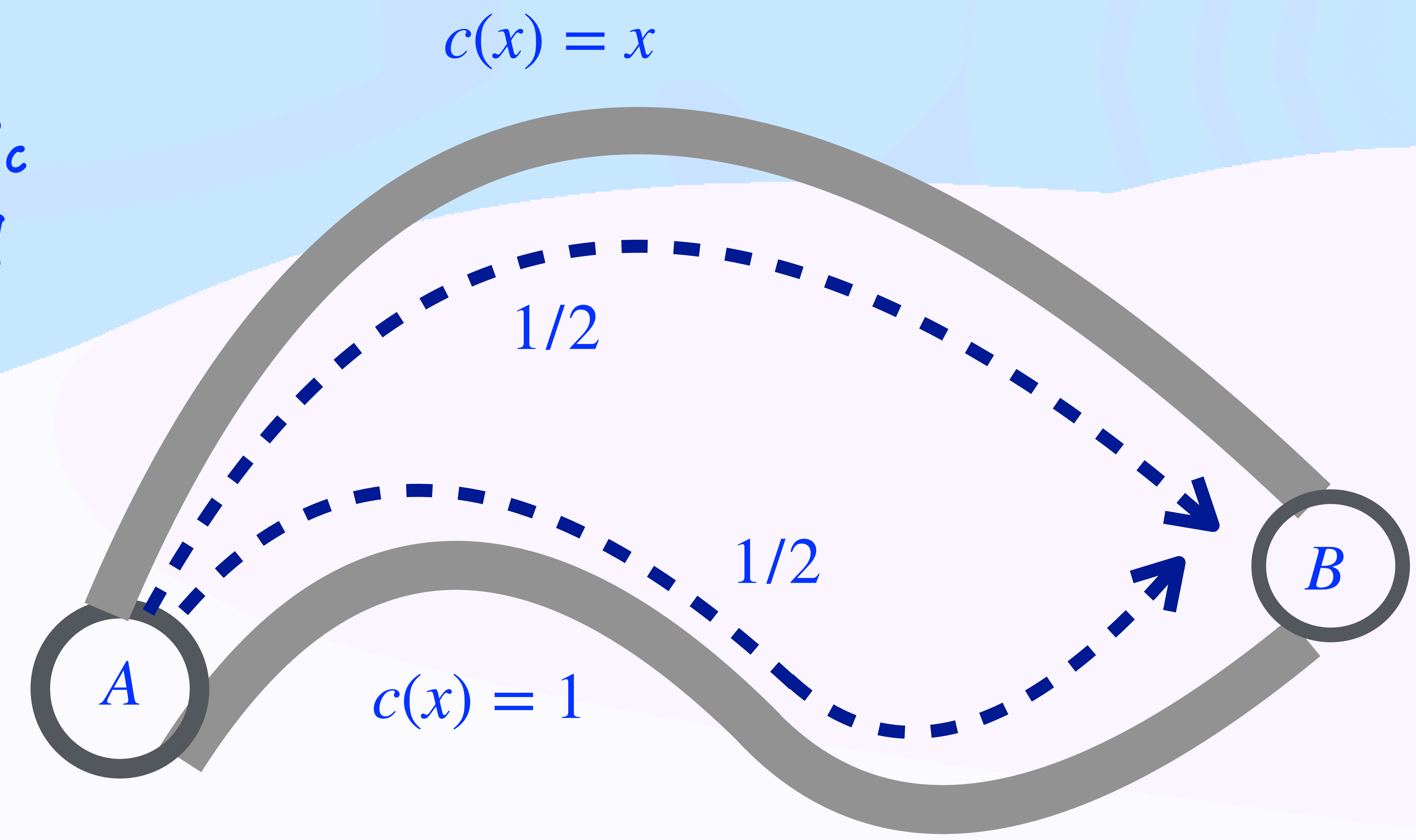
Pigou's Network 1920

1 unit of traffic

Dominant Strategy is



Traffic control



Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

Expected travel time $= \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$

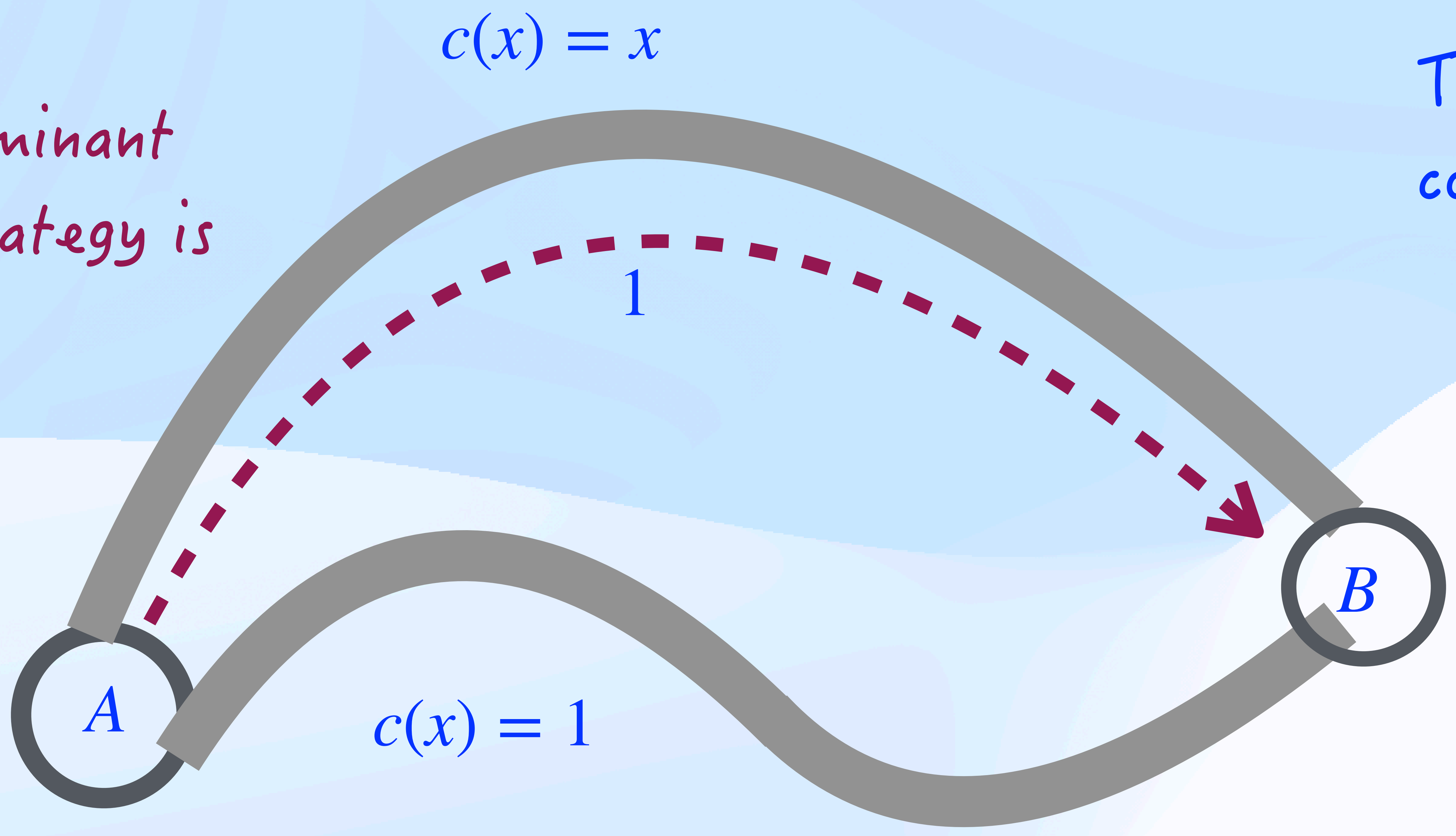
Q: Is this the ideal outcome ?

Let's try to understand this phenomenon

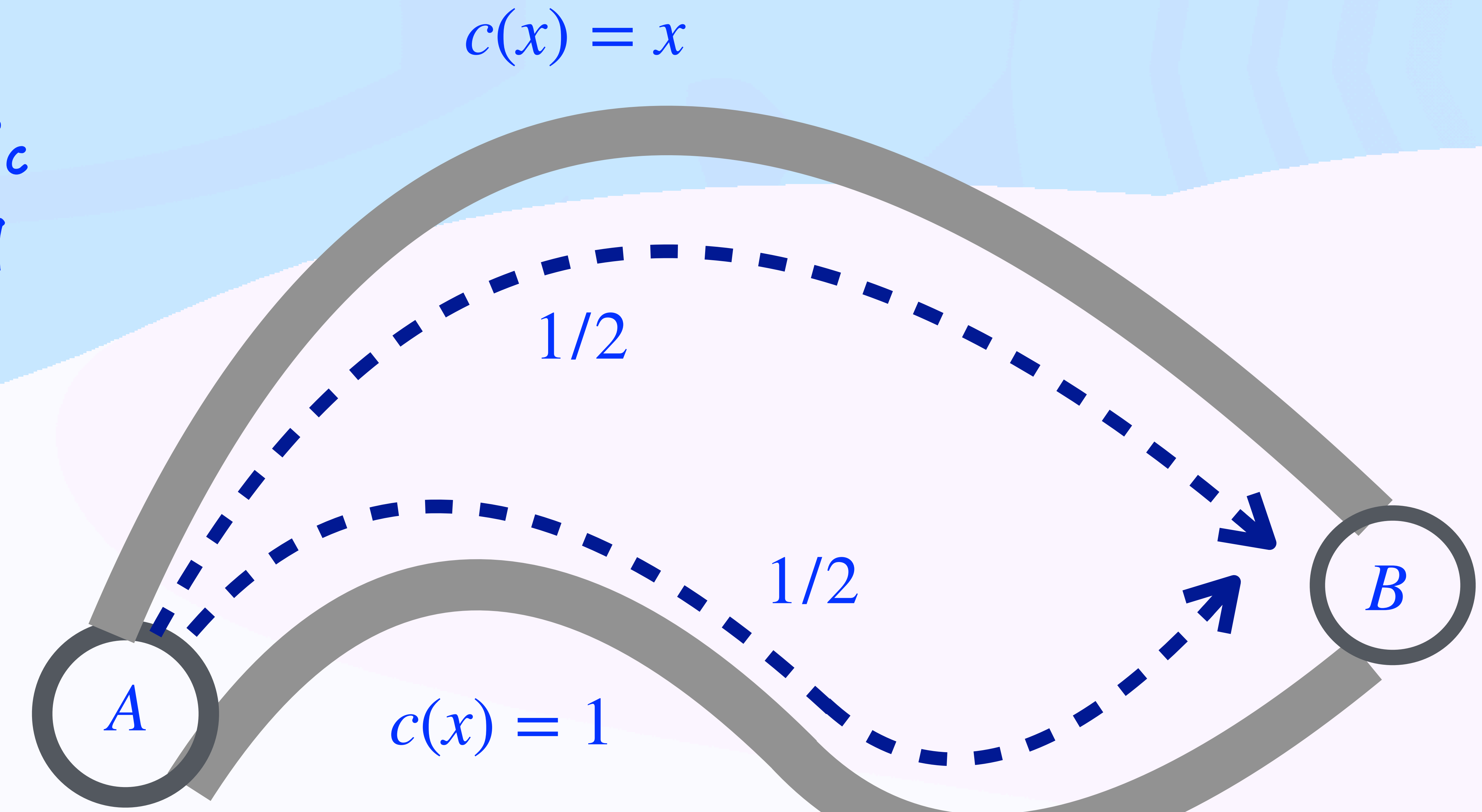
Pigou's Network 1920

1 unit of traffic

Dominant Strategy is



Traffic control



Expected travel time in DS equilibria $= 1 \cdot 1 = 1$

Expected travel time $= \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$

Q: Is this the ideal outcome ?

$$\frac{\text{Travel time in DS equilibrium}}{\text{Min average travel time}} = \frac{1}{3/4} = \frac{4}{3}$$

Q: What should be our social objective ?

Q: What should be our social objective ?

Ans: Minimize average travel time

Q: What should be our social objective ?

Ans: Minimize average travel time

Q: How far can we be from this objective ?

Q: What should be our social objective ?

Ans: Minimize average travel time

Q: How far can we be from this objective ?

Price of Anarchy (POA)

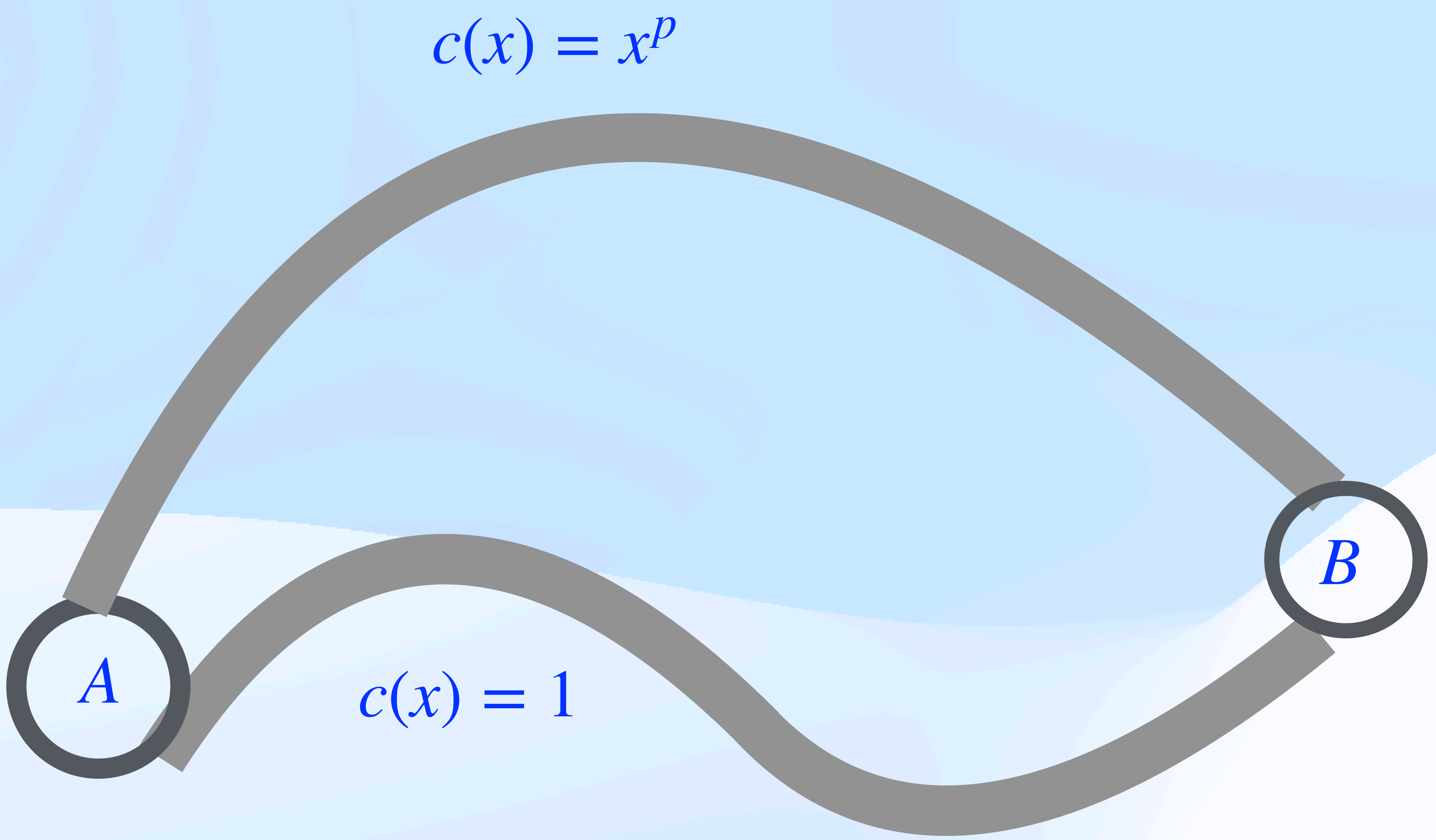
Q: What should be our social objective ?

Ans: Minimize average travel time

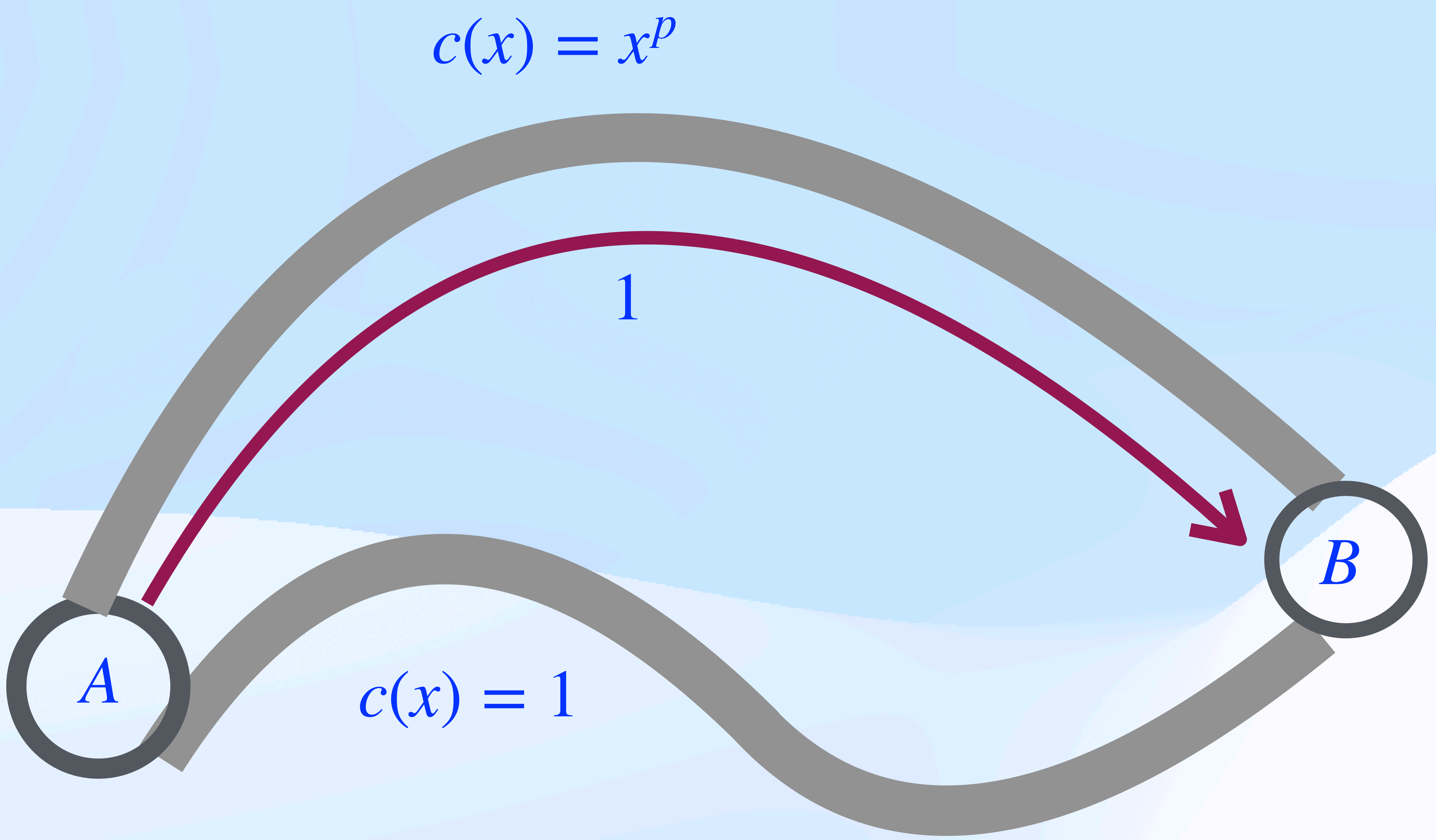
Q: How far can we be from this objective ?

$$\text{Price of Anarchy (POA)} = \frac{\text{Travel time in DS equilibrium}}{\text{Min possible average travel time}}$$

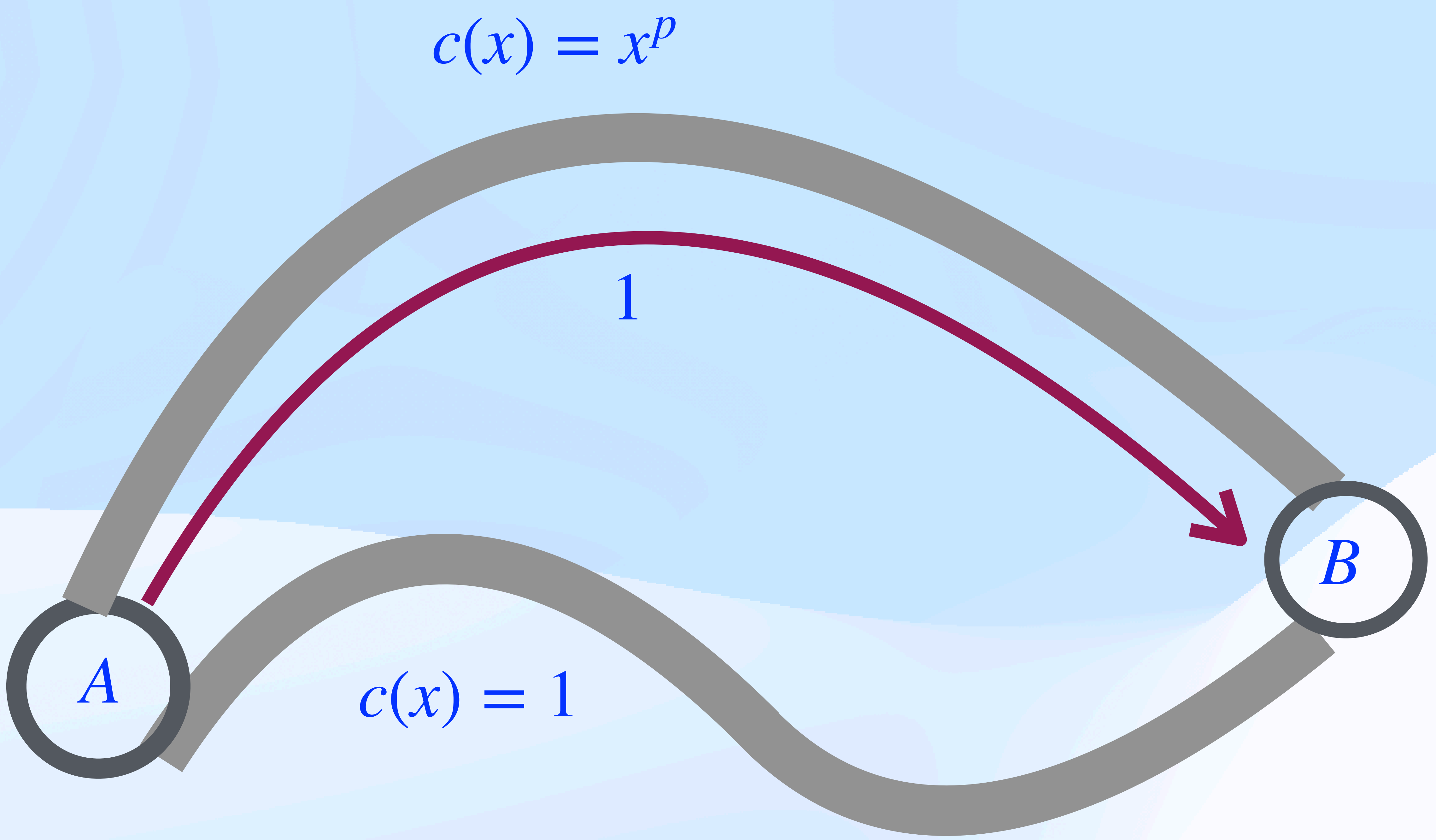
What if the cost is non-linear ?



What if the cost is non-linear ?

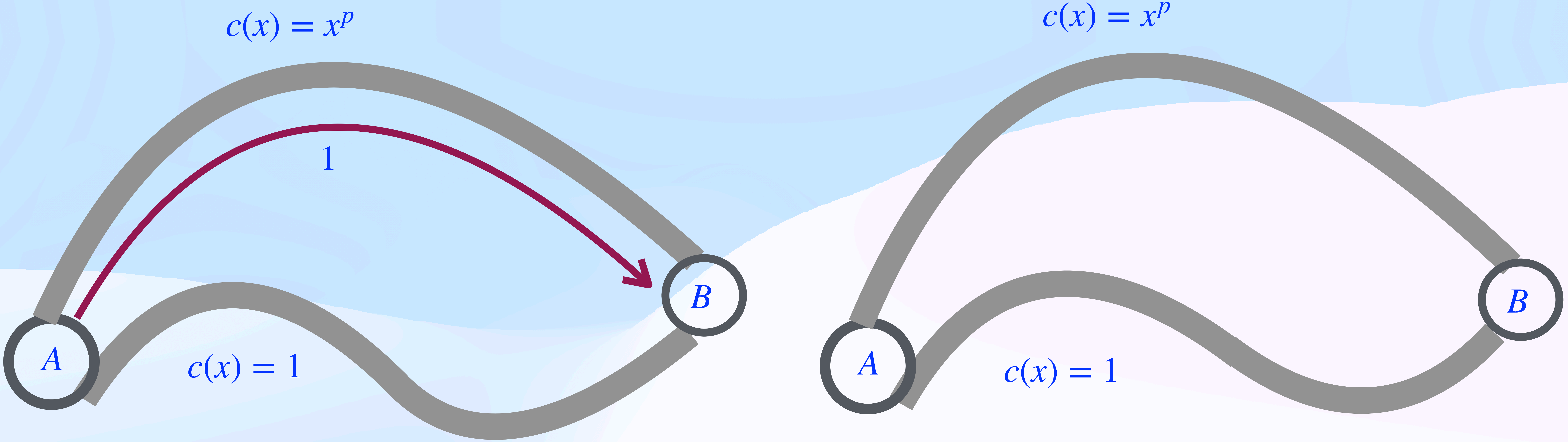


What if the cost is non-linear ?



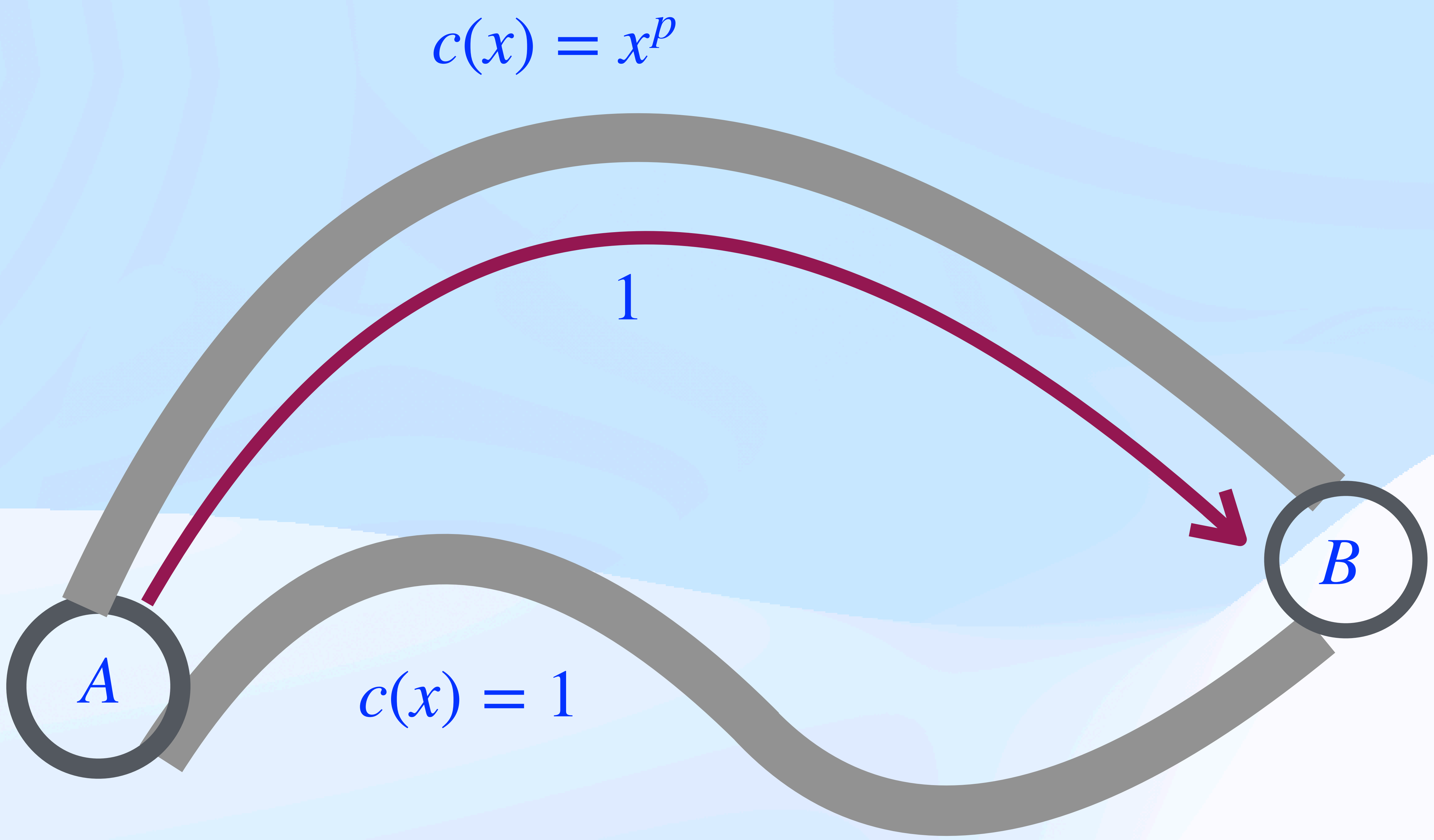
Travel time in DS equilibria $= 1$

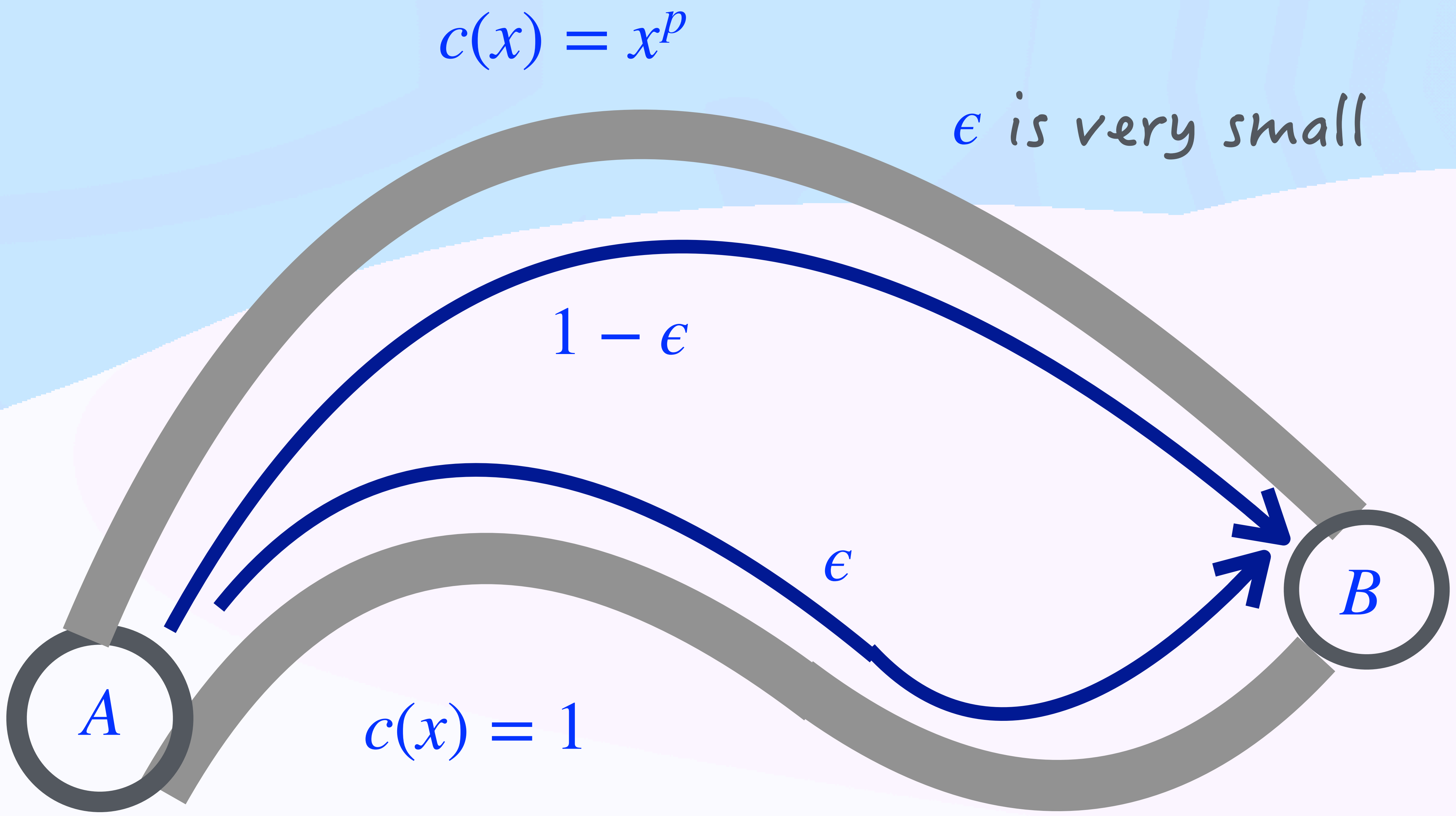
What if the cost is non-linear ?



Travel time in DS equilibria $= 1$

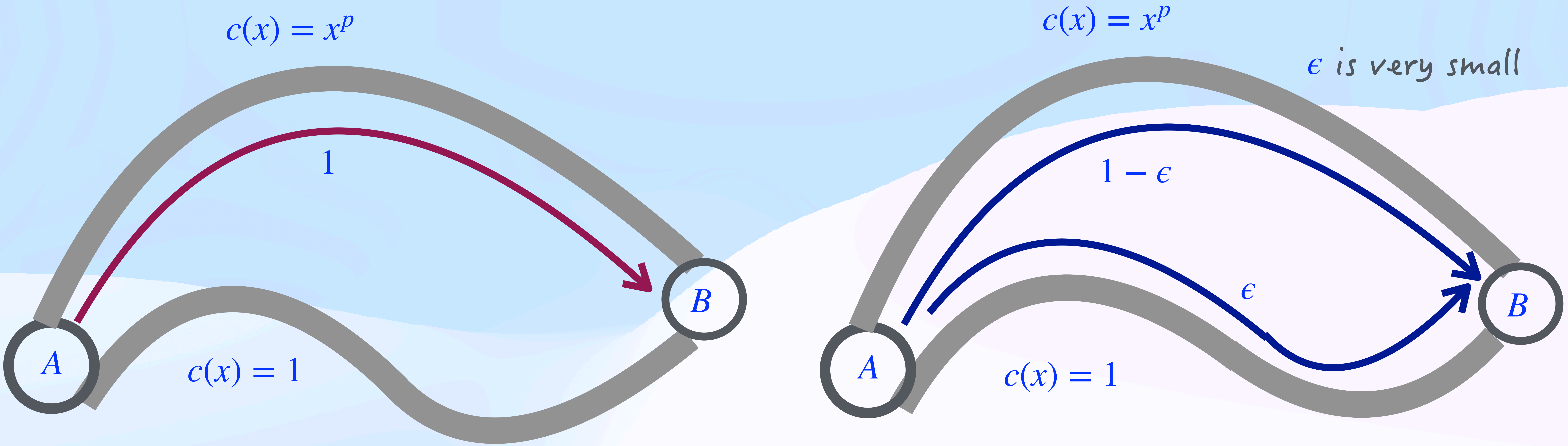
What if the cost is non-linear ?





Travel time in DS equilibria $= 1$

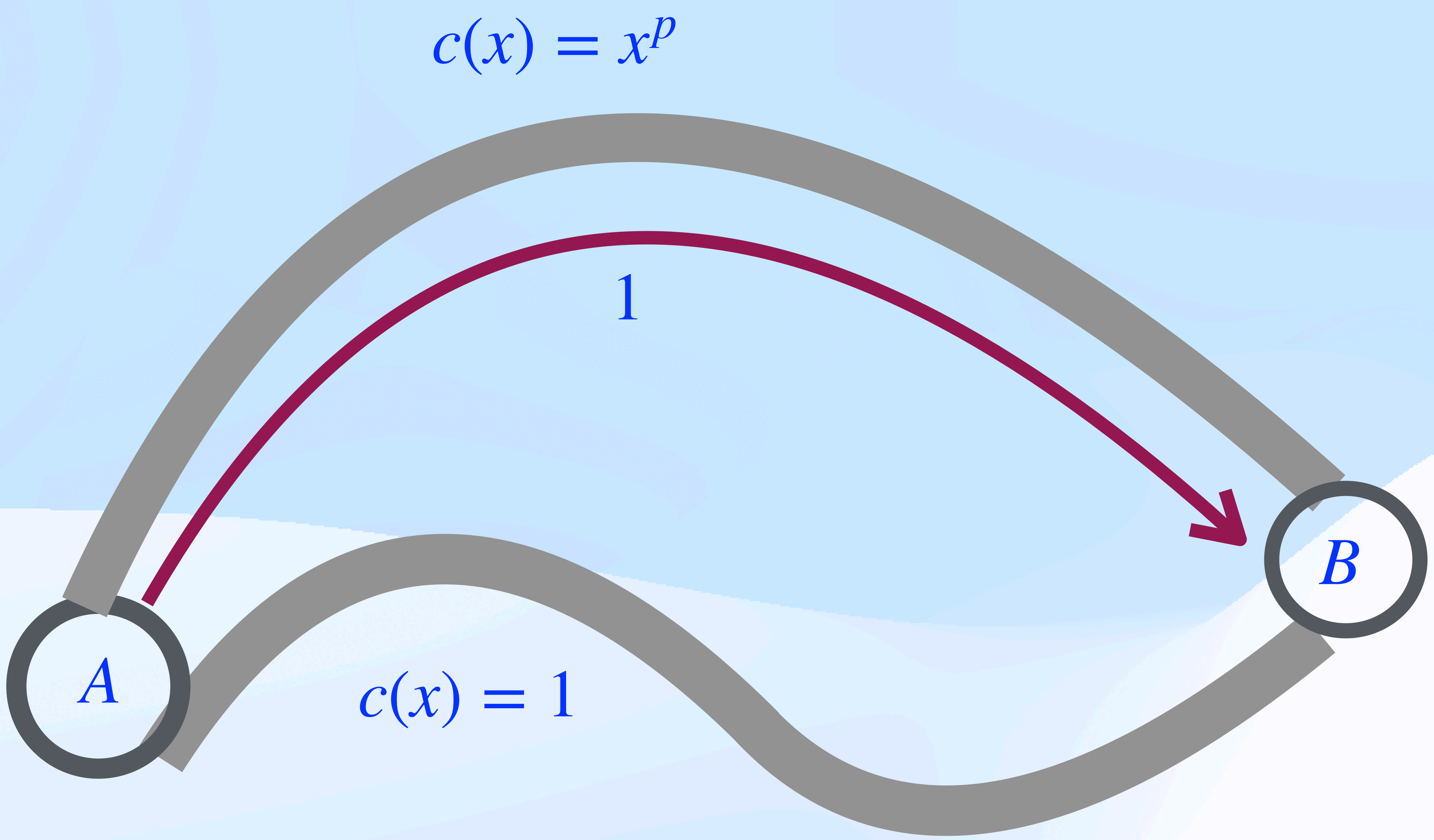
What if the cost is non-linear ?

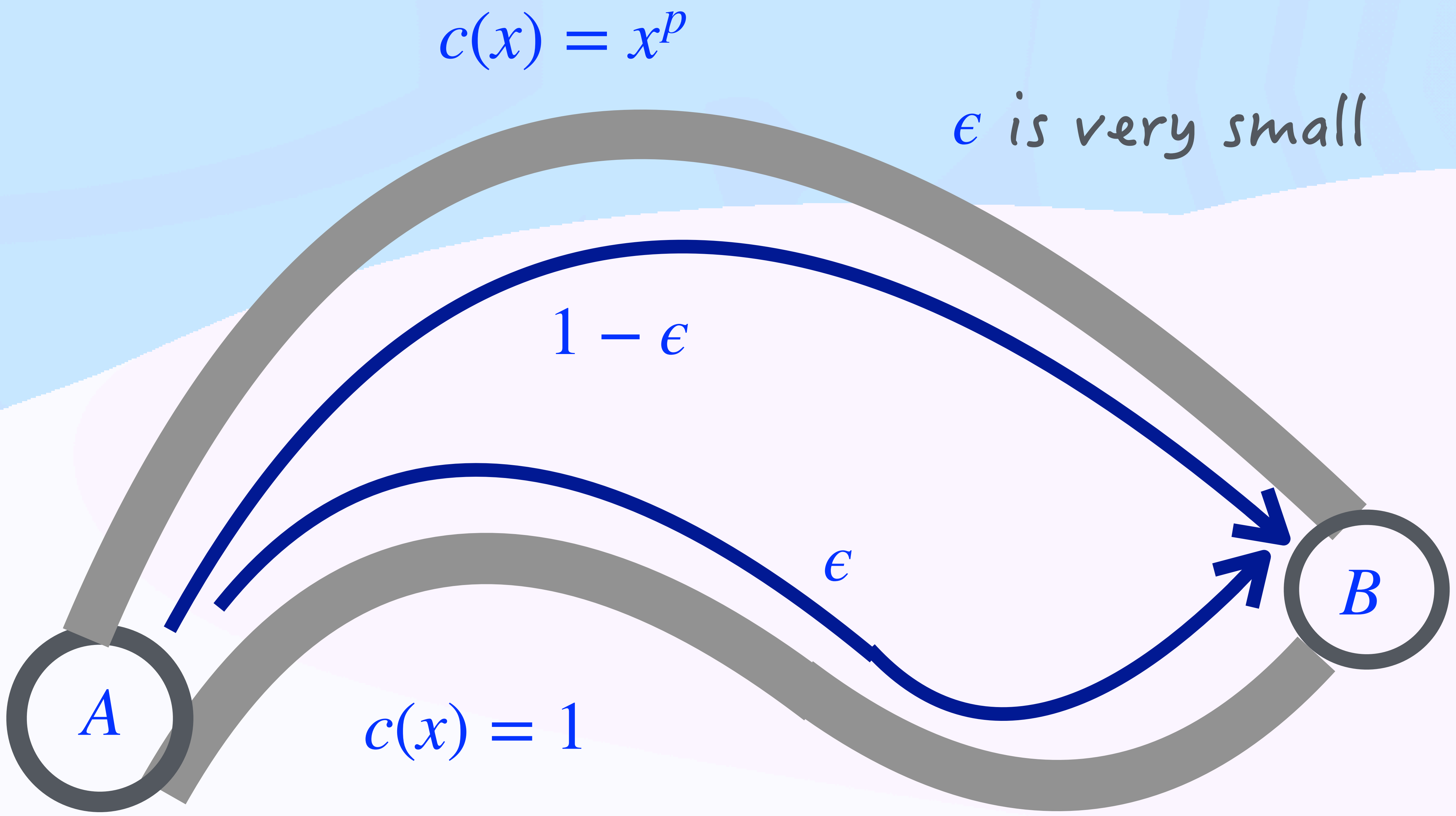


Travel time in DS equilibria $= 1$

Travel time

What if the cost is non-linear ?



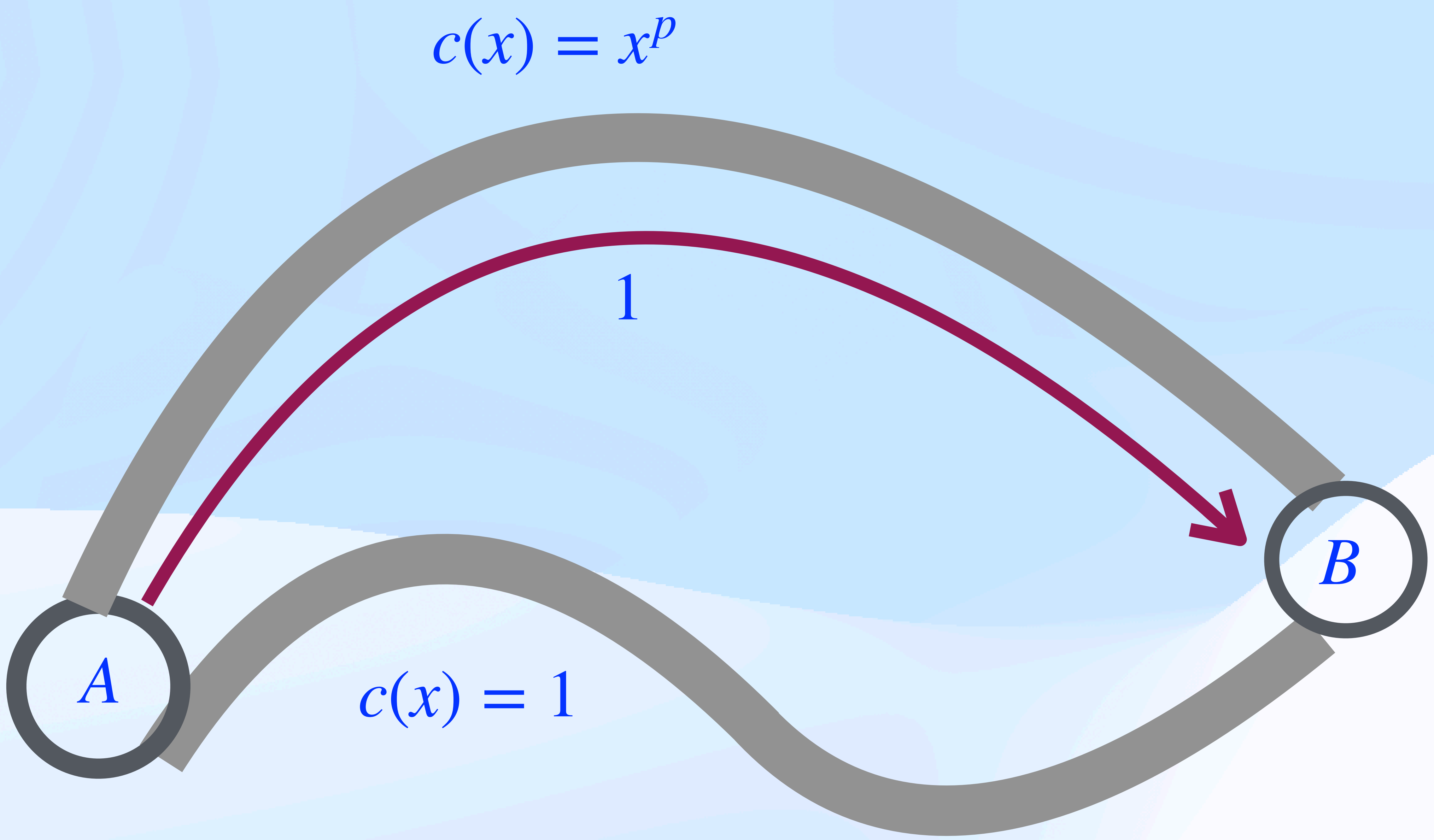


Travel time in DS equilibria $= 1$

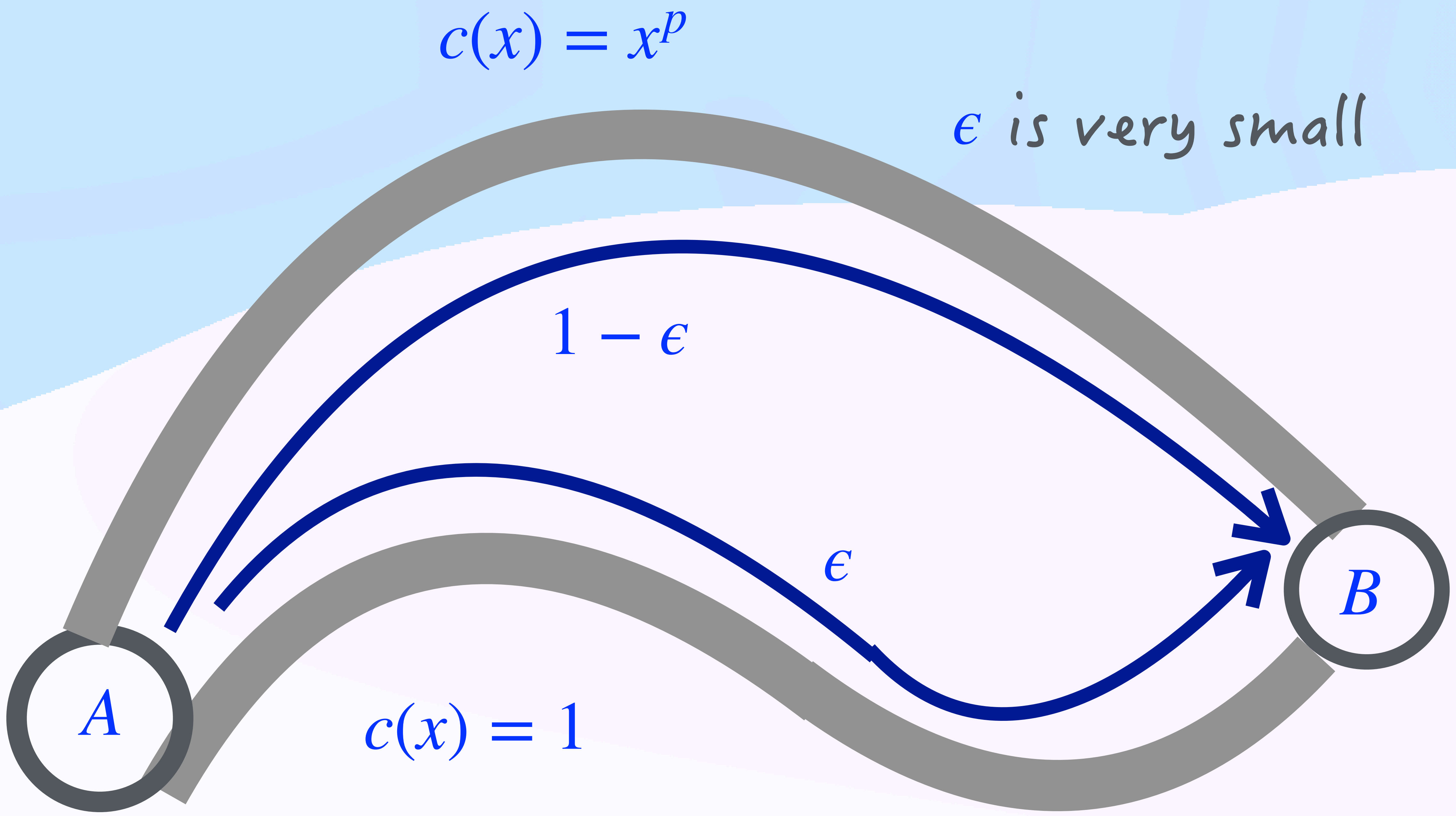
Travel time

$$= \epsilon \cdot 1 + (1-\epsilon)\cdot(1-\epsilon)^p = \epsilon + (1-\epsilon)^{p+1}$$

What if the cost is non-linear ?



Travel time in DS equilibria $= 1$

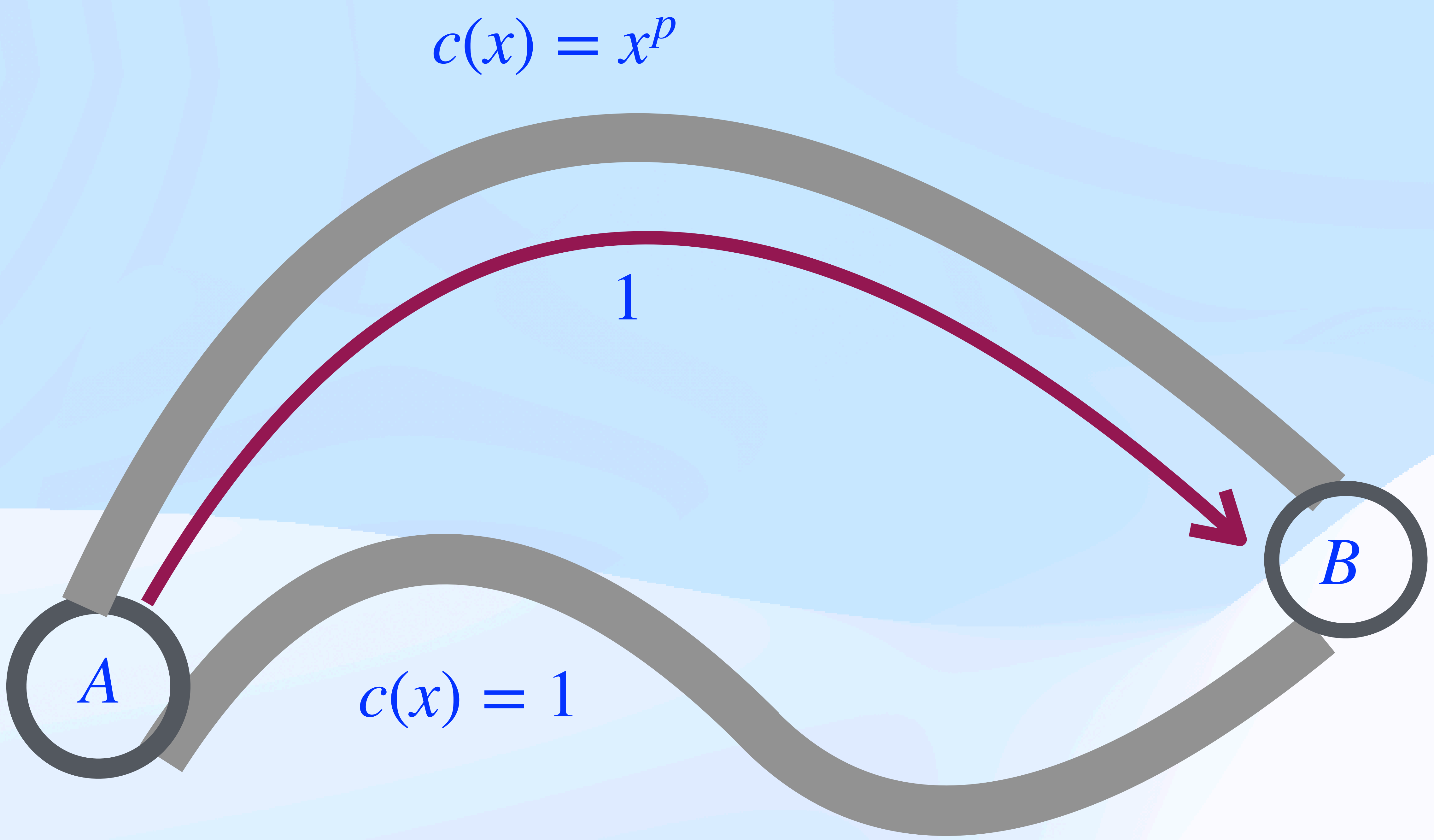


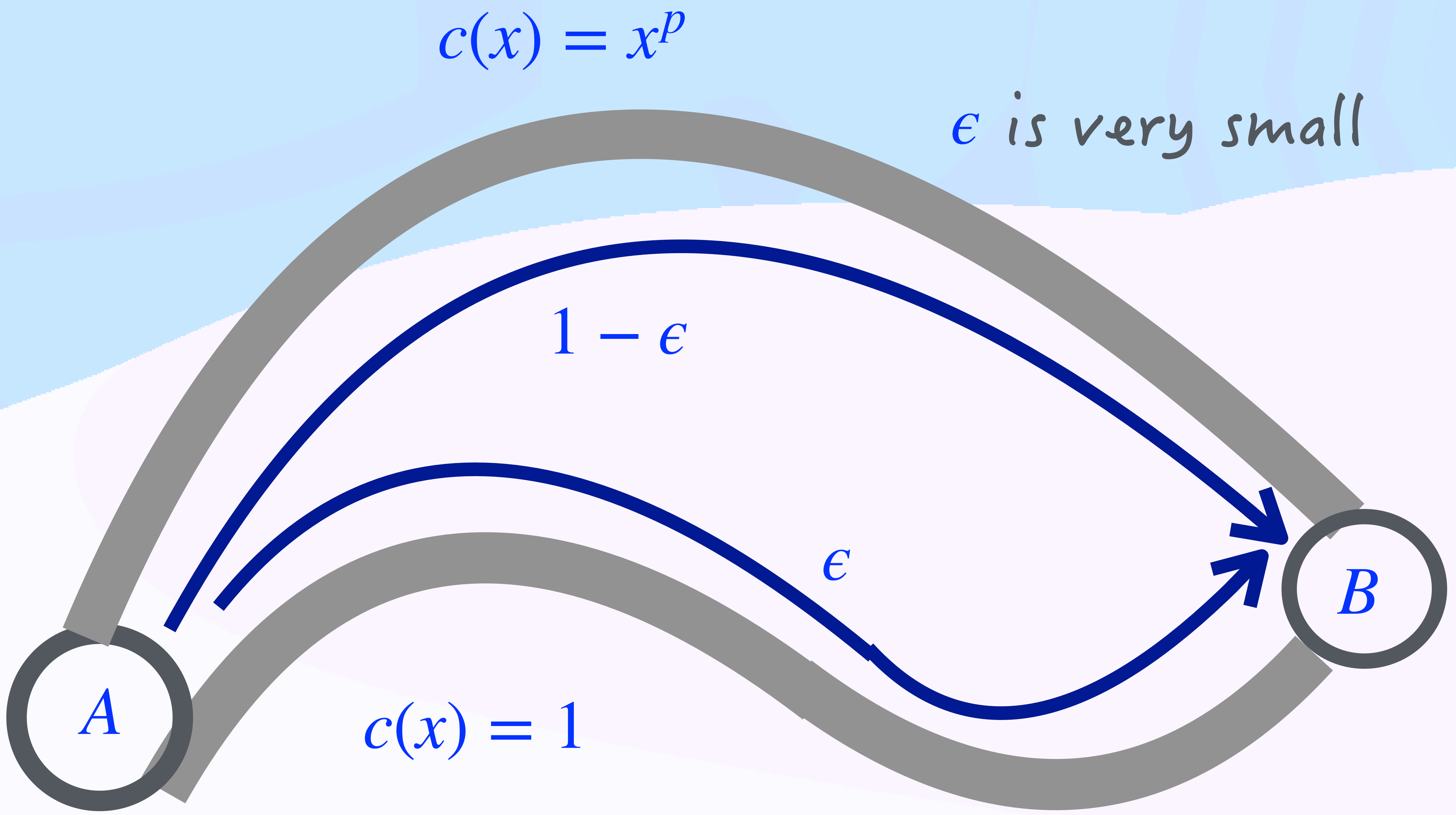
Travel time

$$= \epsilon \cdot 1 + (1 - \epsilon) \cdot (1 - \epsilon)^p = \epsilon + (1 - \epsilon)^{p+1}$$

$\to 0$ when p is large enough

What if the cost is non-linear ?





ϵ is very small

Travel time in DS equilibria $= 1$

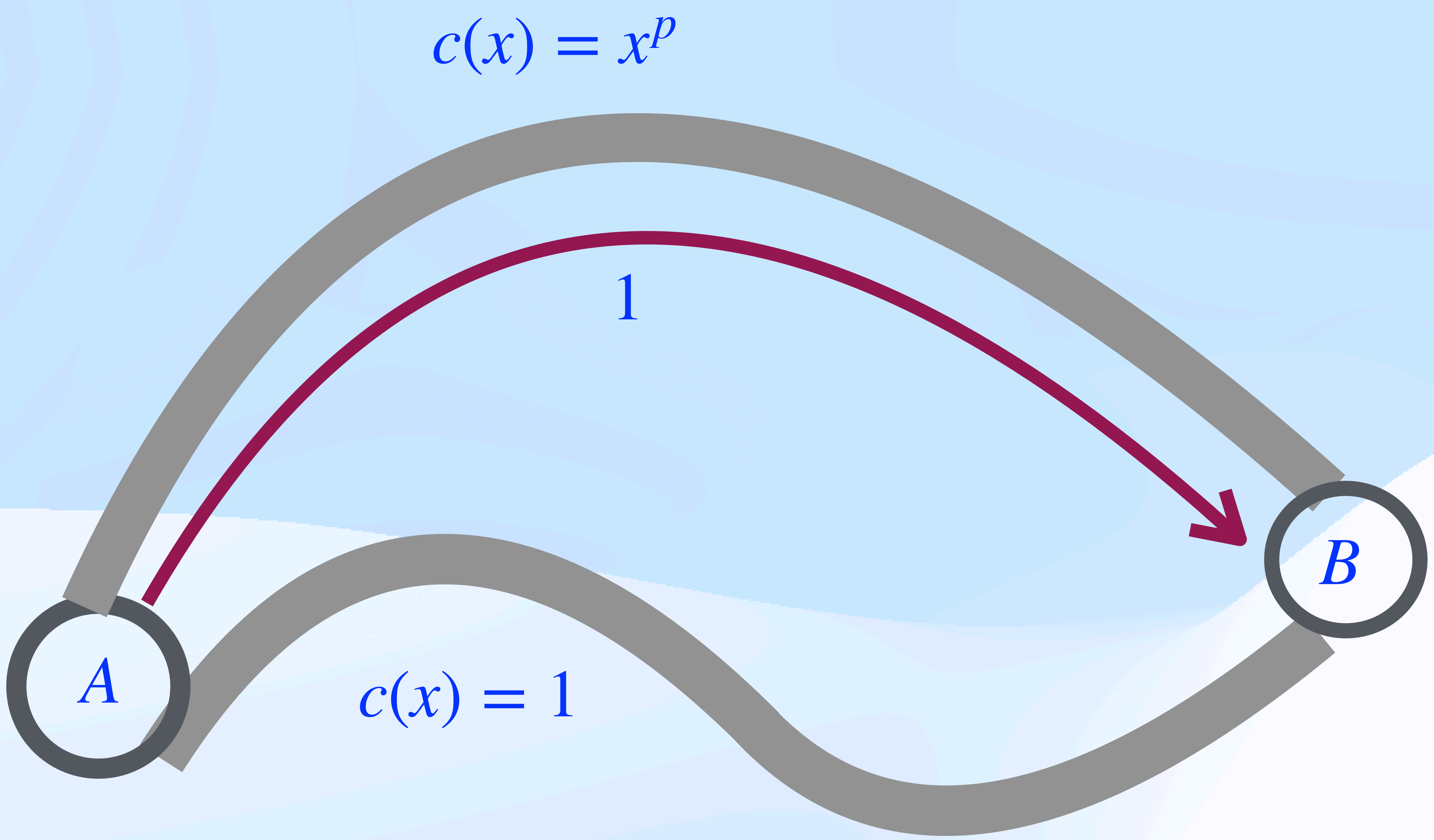
Travel time

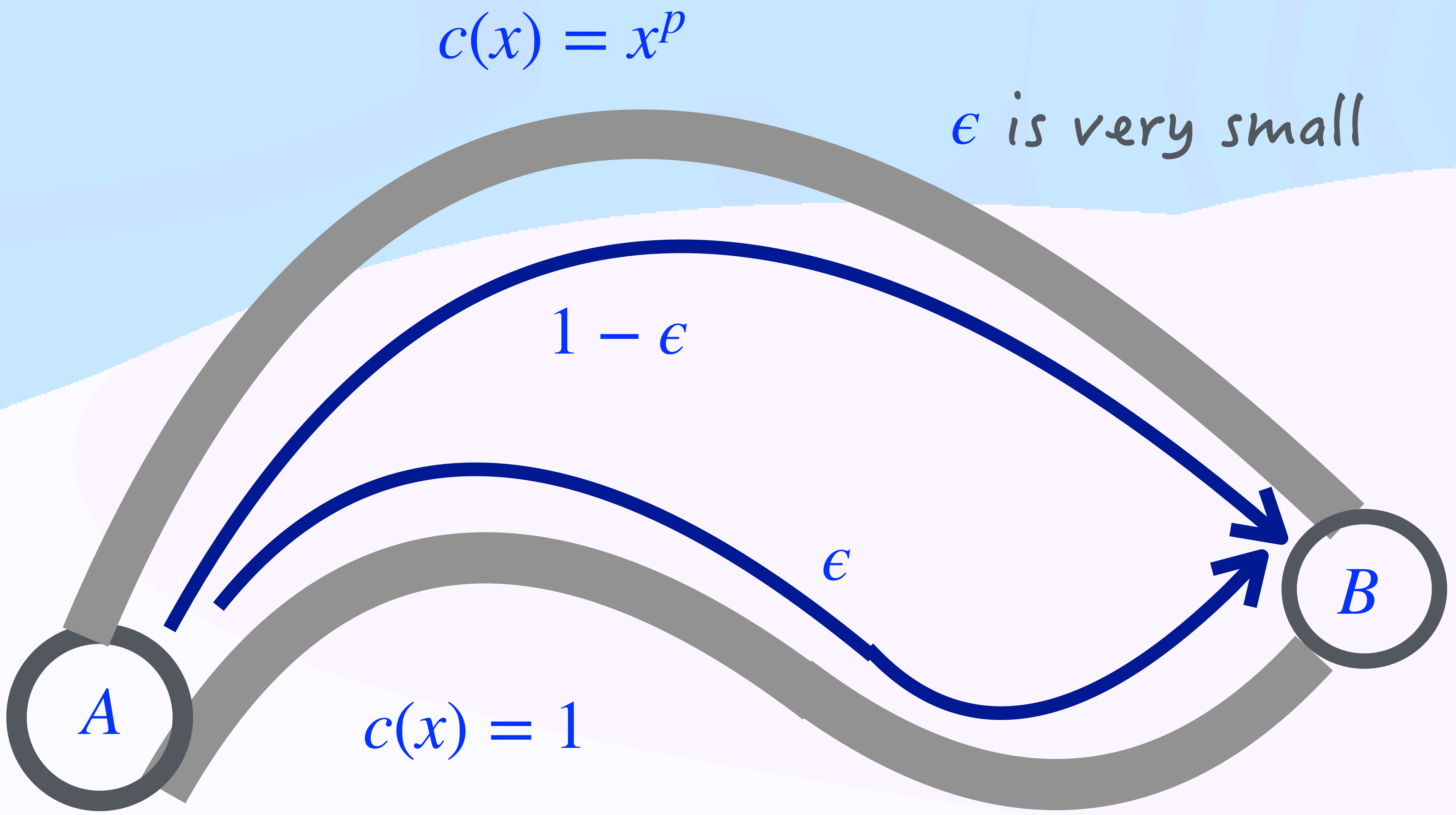
$$= \epsilon \cdot 1 + (1-\epsilon)\cdot(1-\epsilon)^p = \epsilon + (1-\epsilon)^{p+1}$$

$\to 0$ when p is large enough

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}}$$

What if the cost is non-linear ?





Travel time in DS equilibria $= 1$

Travel time

$$= \epsilon \cdot 1 + (1 - \epsilon) \cdot (1 - \epsilon)^p = \epsilon + (1 - \epsilon)^{p+1}$$

$\rightarrow 0$ when p is large enough

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}}$$

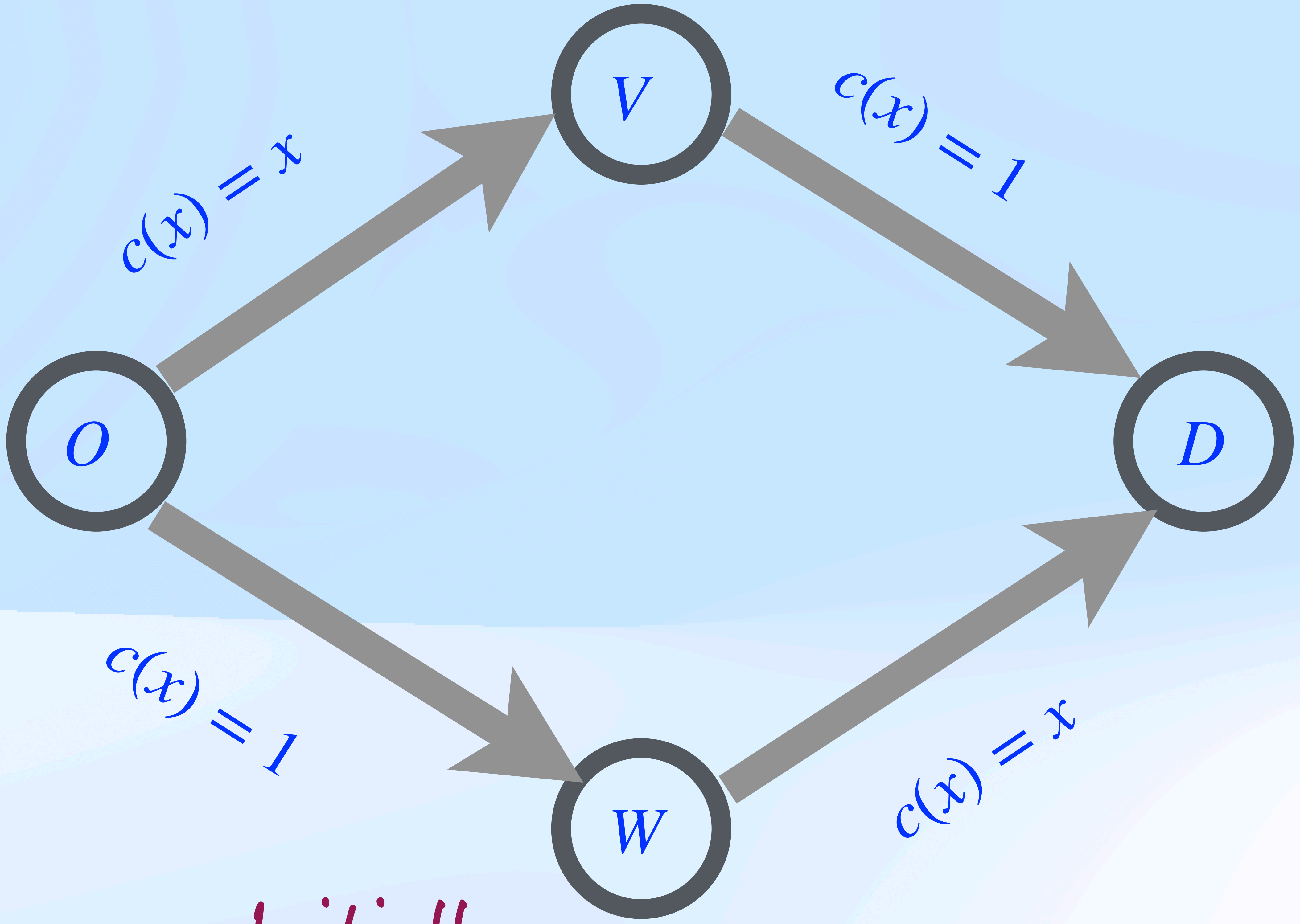
Is unbounded!

Braess Network, 1965

Initially

Braess Network, 1965

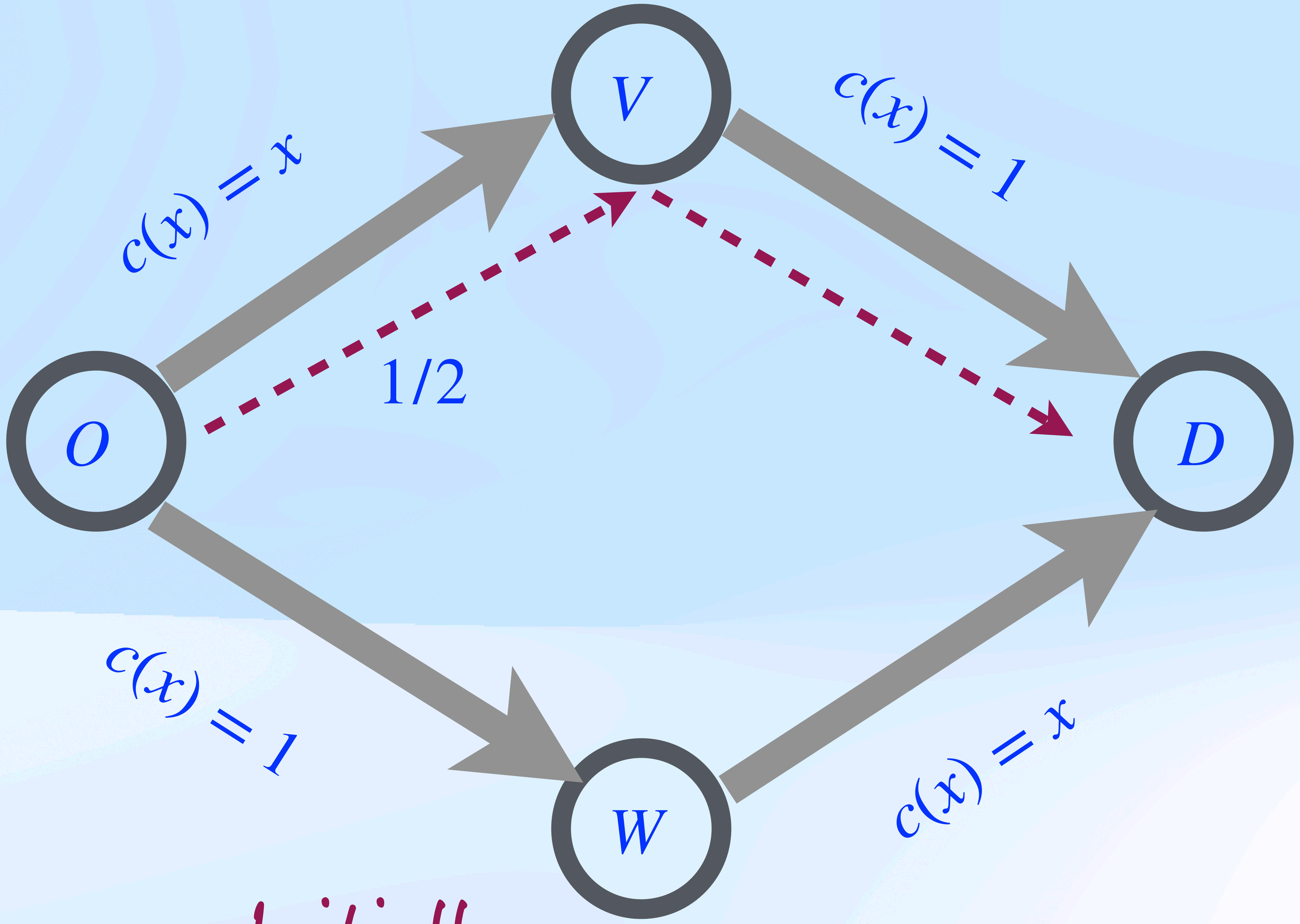
1 unit of traffic



Initially

Braess Network, 1965

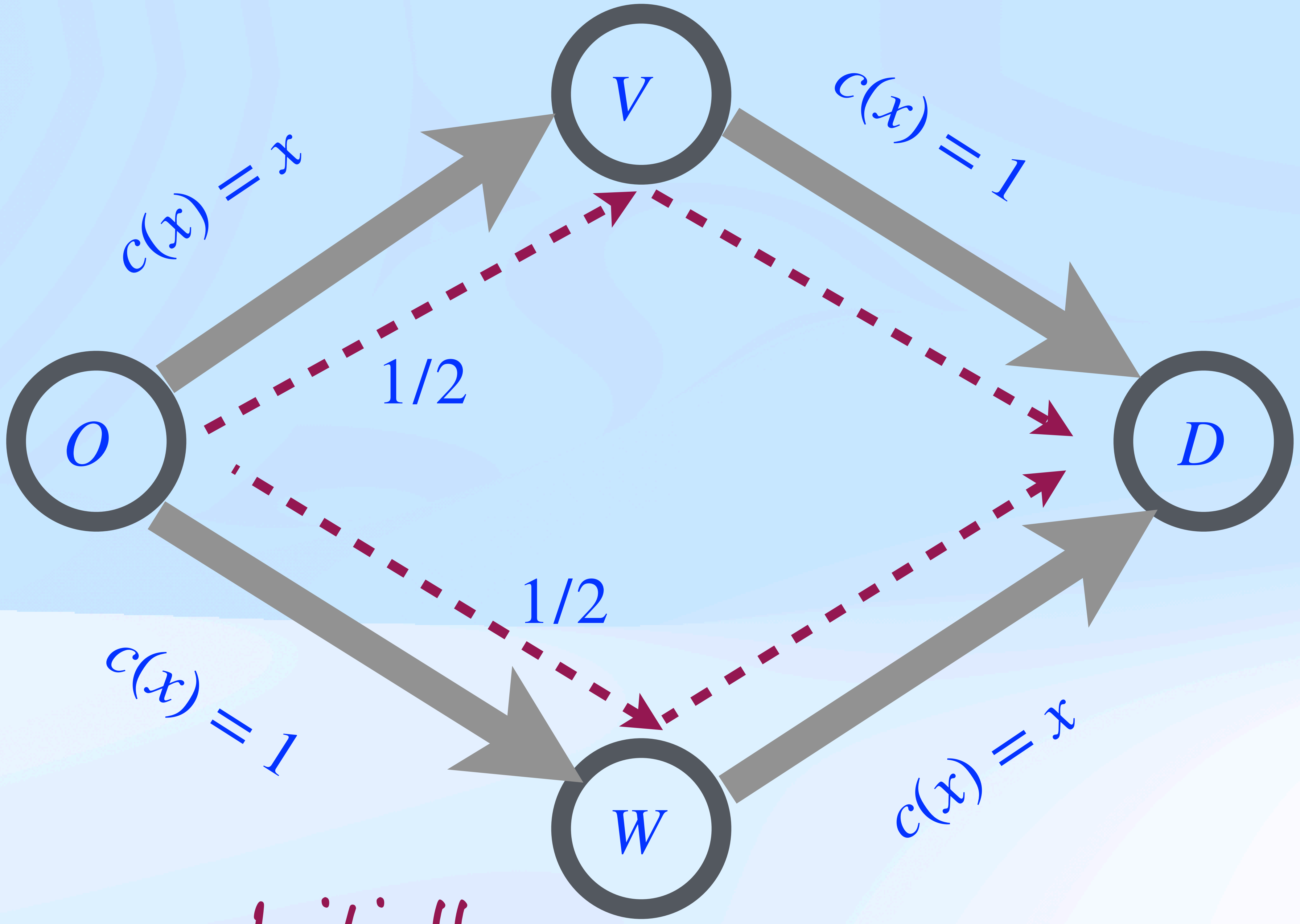
1 unit of traffic



Initially

Braess Network, 1965

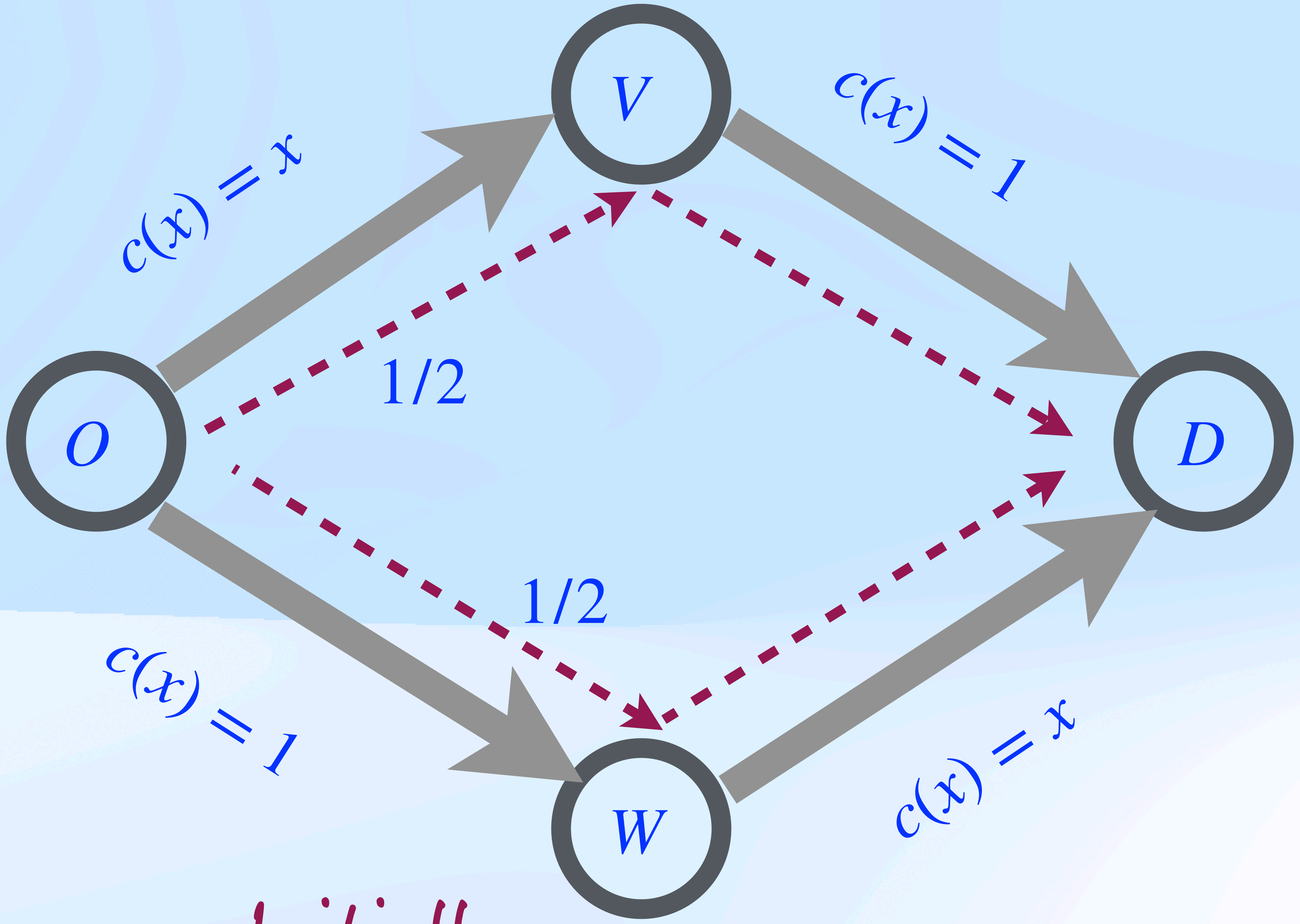
1 unit of traffic



Initially

Braess Network, 1965

1 unit of traffic

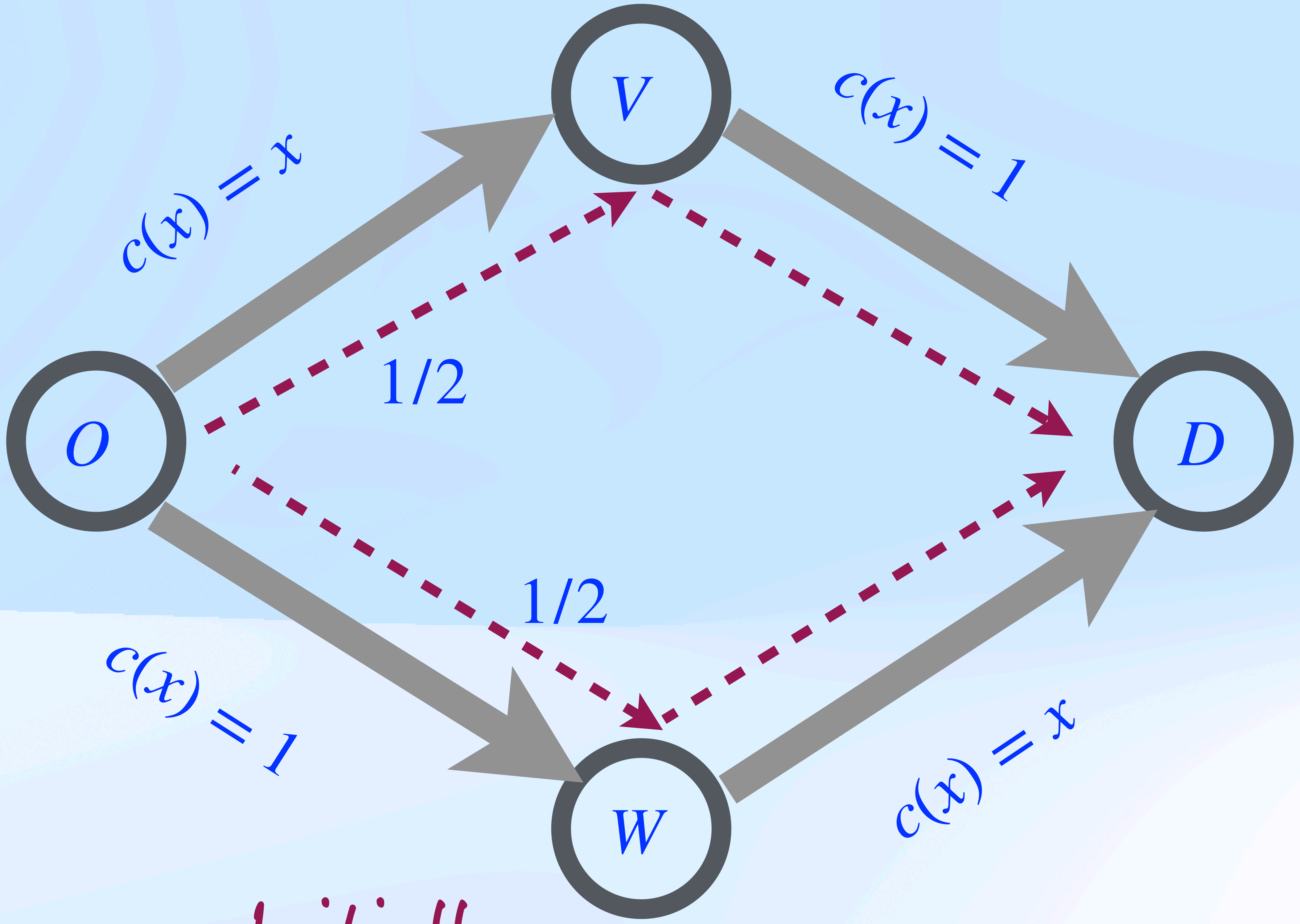


Initially

Travel time

Braess Network, 1965

1 unit of traffic

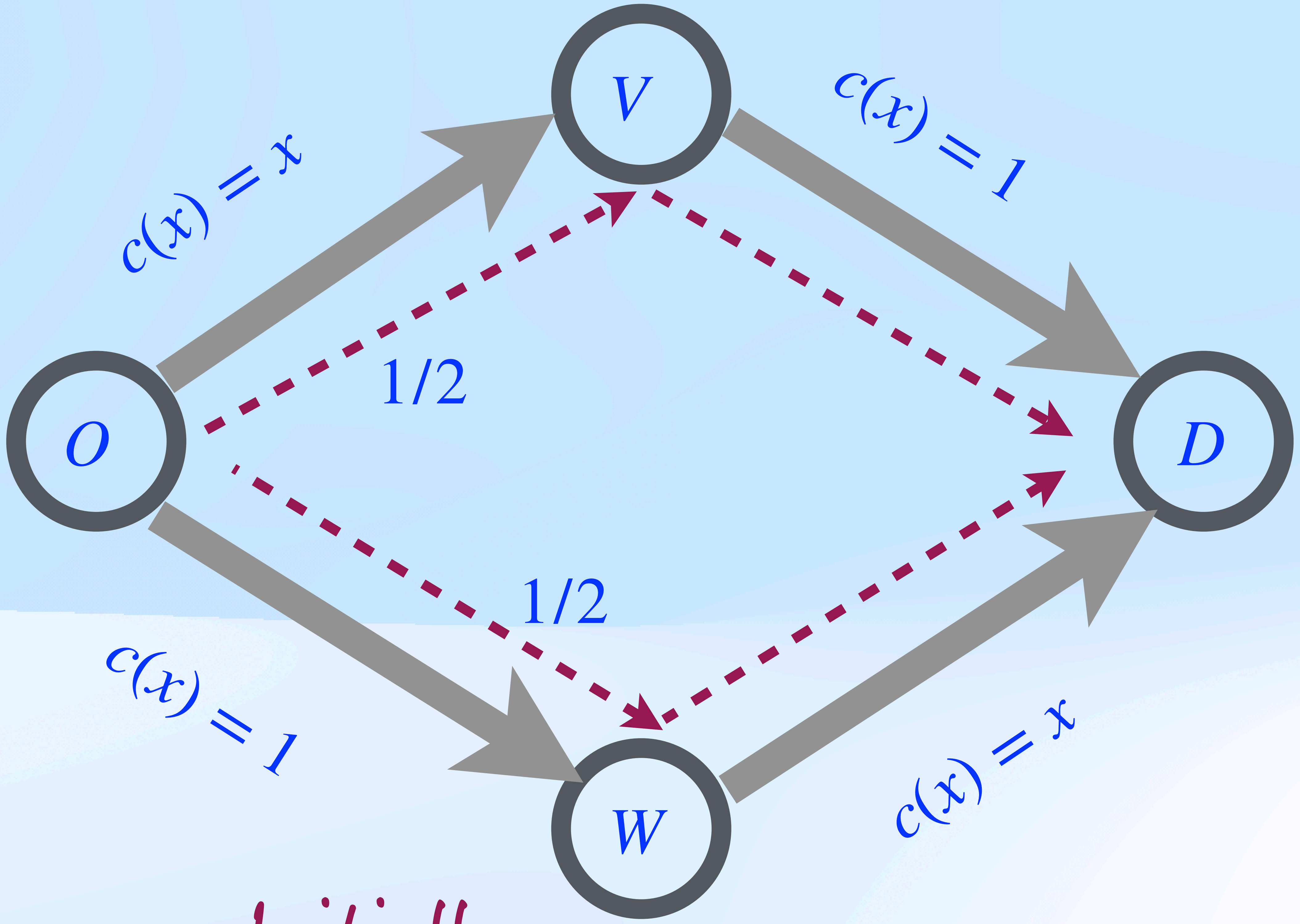


Initially

Travel time $= 2\left(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot 1\right) = \frac{3}{2}$

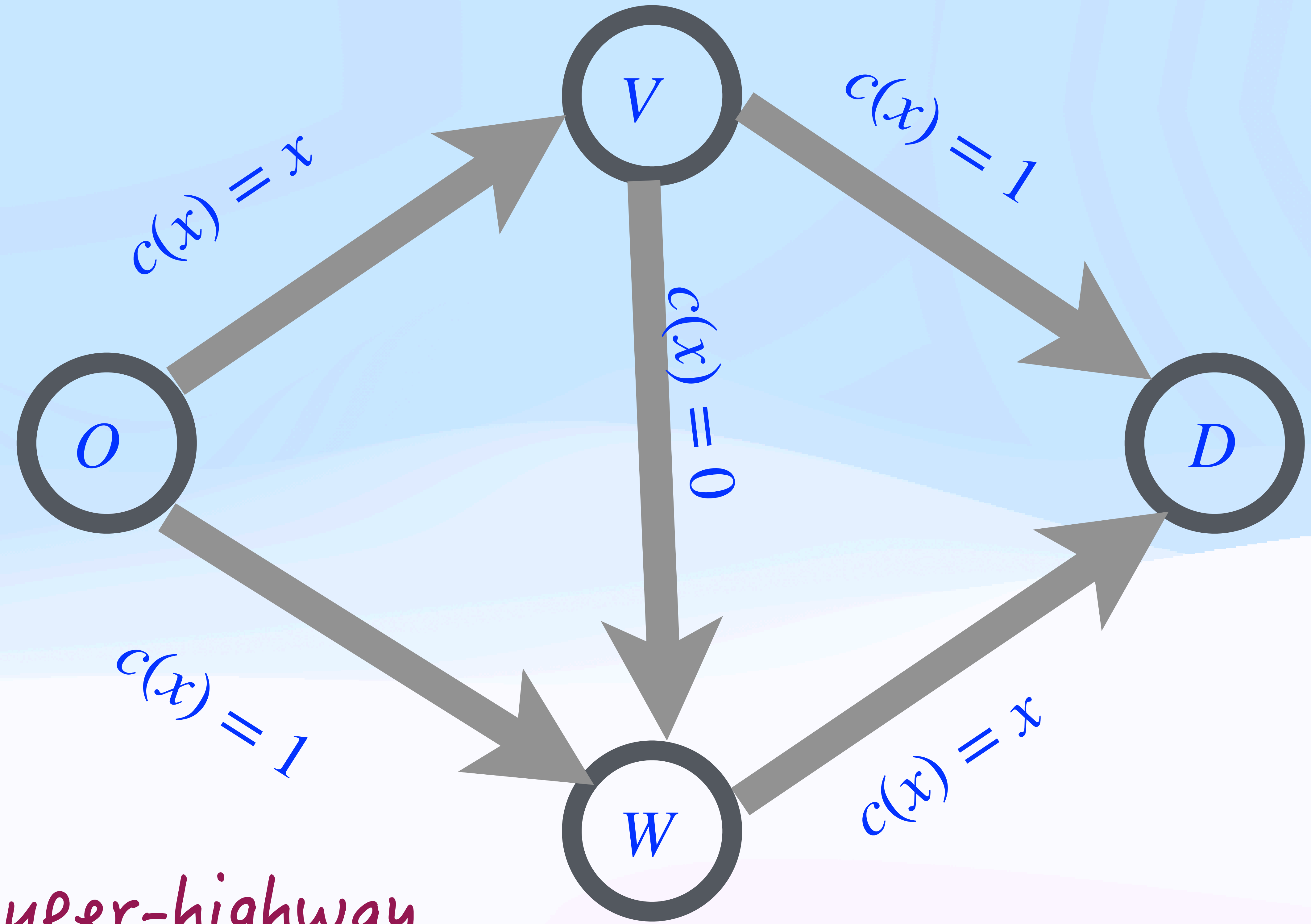
Braess Network, 1965

1 unit of traffic



Initially

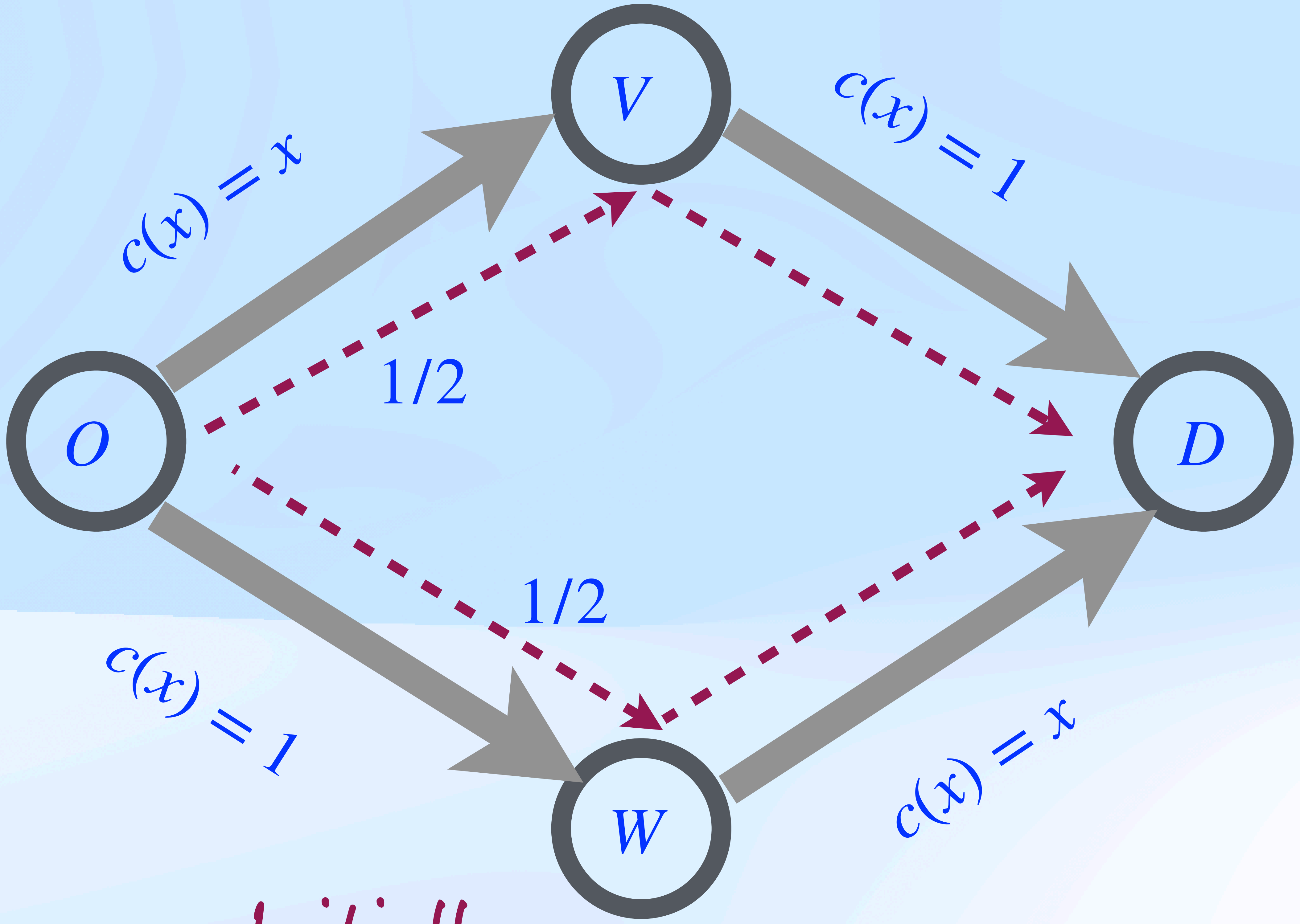
Travel time $= 2\left(\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot 1\right)=\frac{3}{2}$



After super-highway

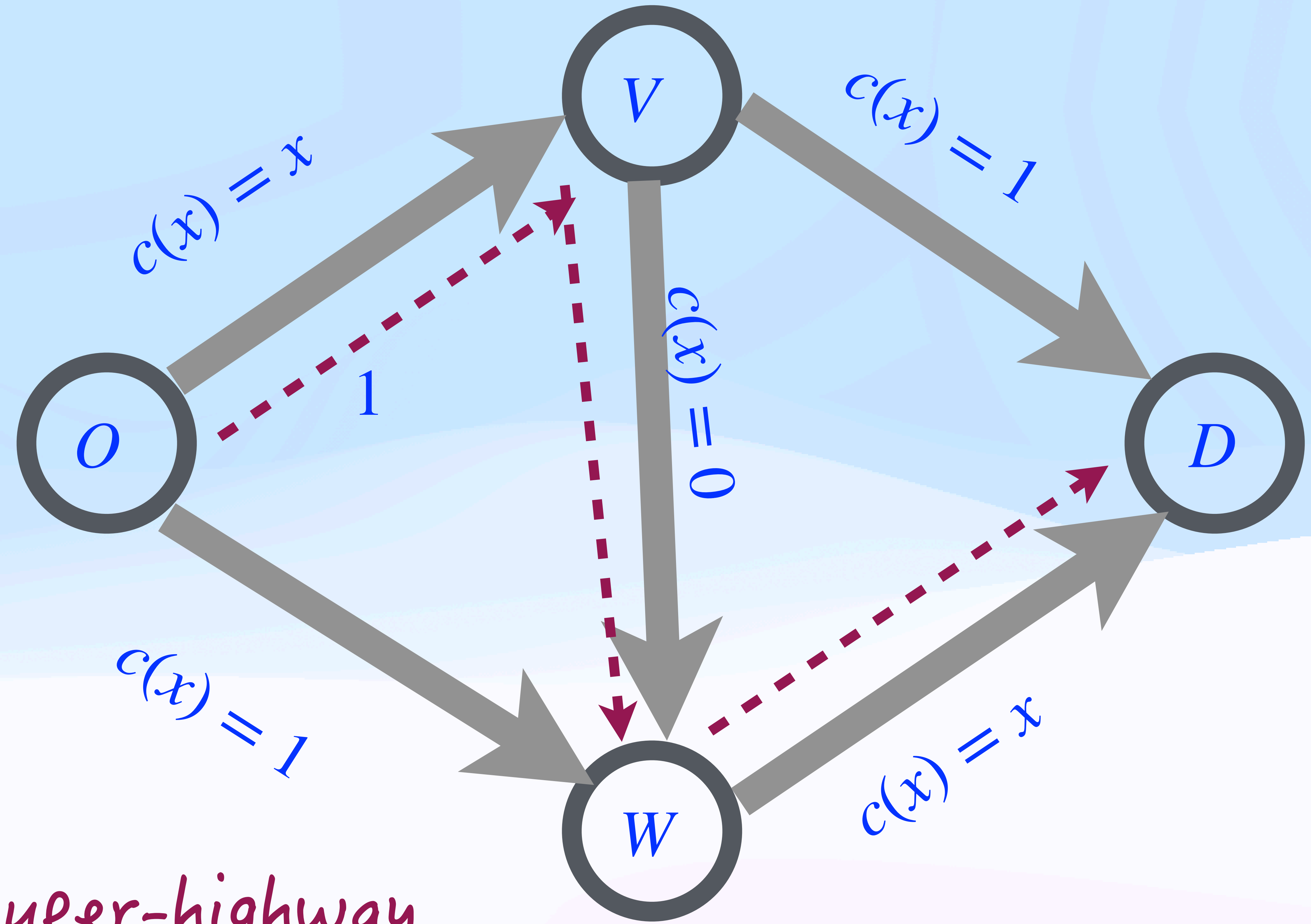
Braess Network, 1965

1 unit of traffic



Initially

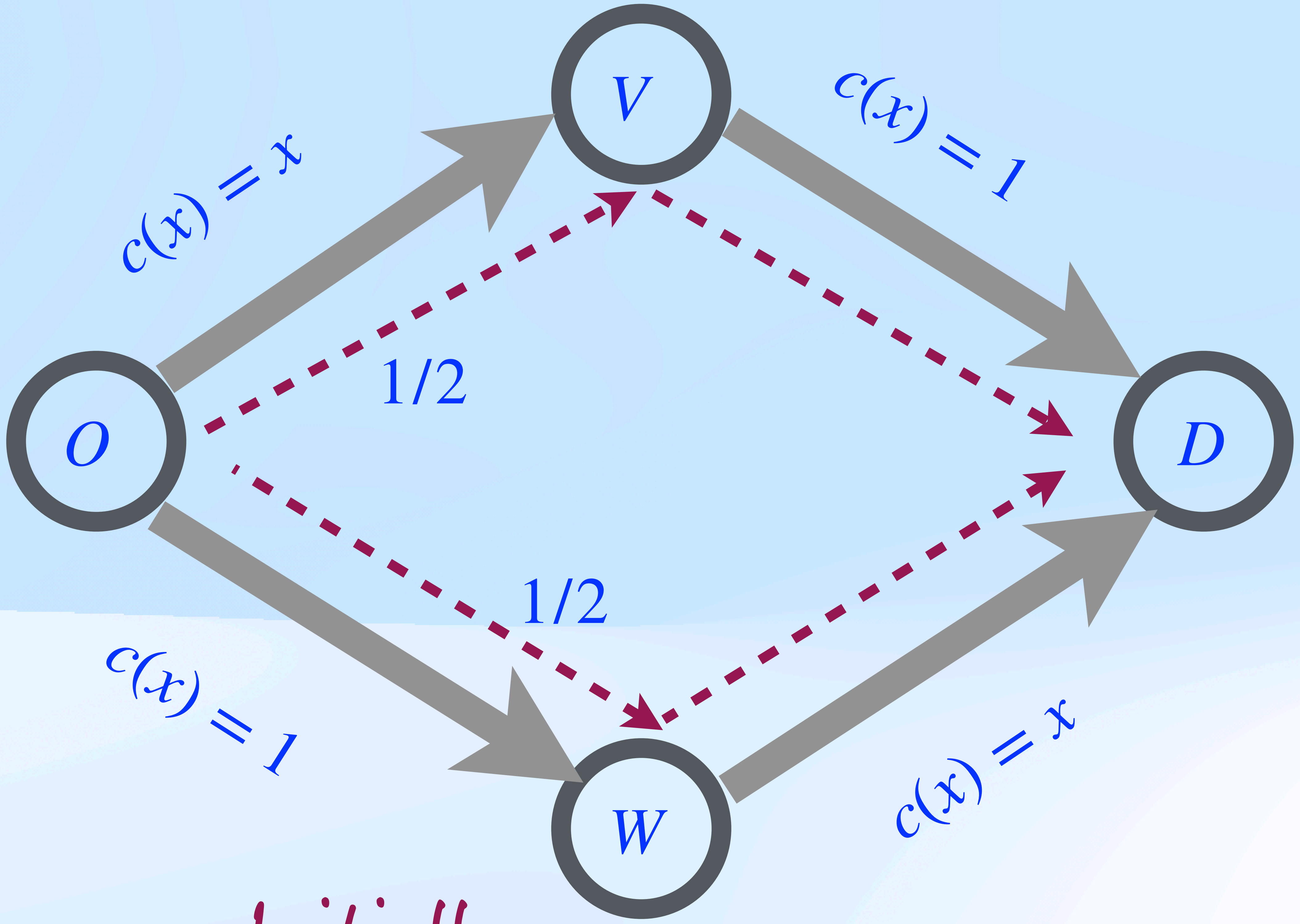
Travel time $= 2\left(\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot 1\right)=\frac{3}{2}$



After super-highway

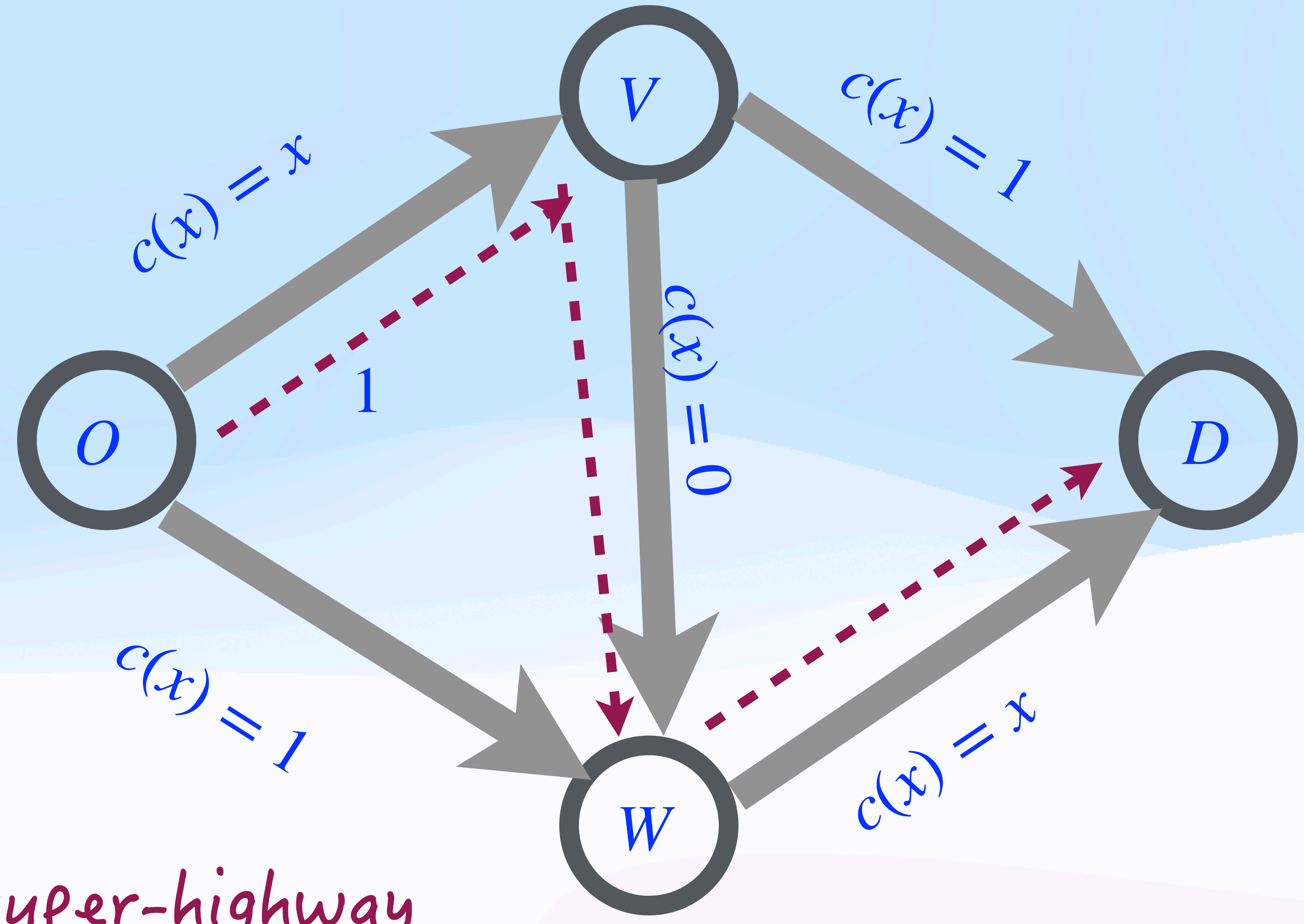
Braess Network, 1965

1 unit of traffic



Initially

Travel time $= 2\left(\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot 1\right)=\frac{3}{2}$

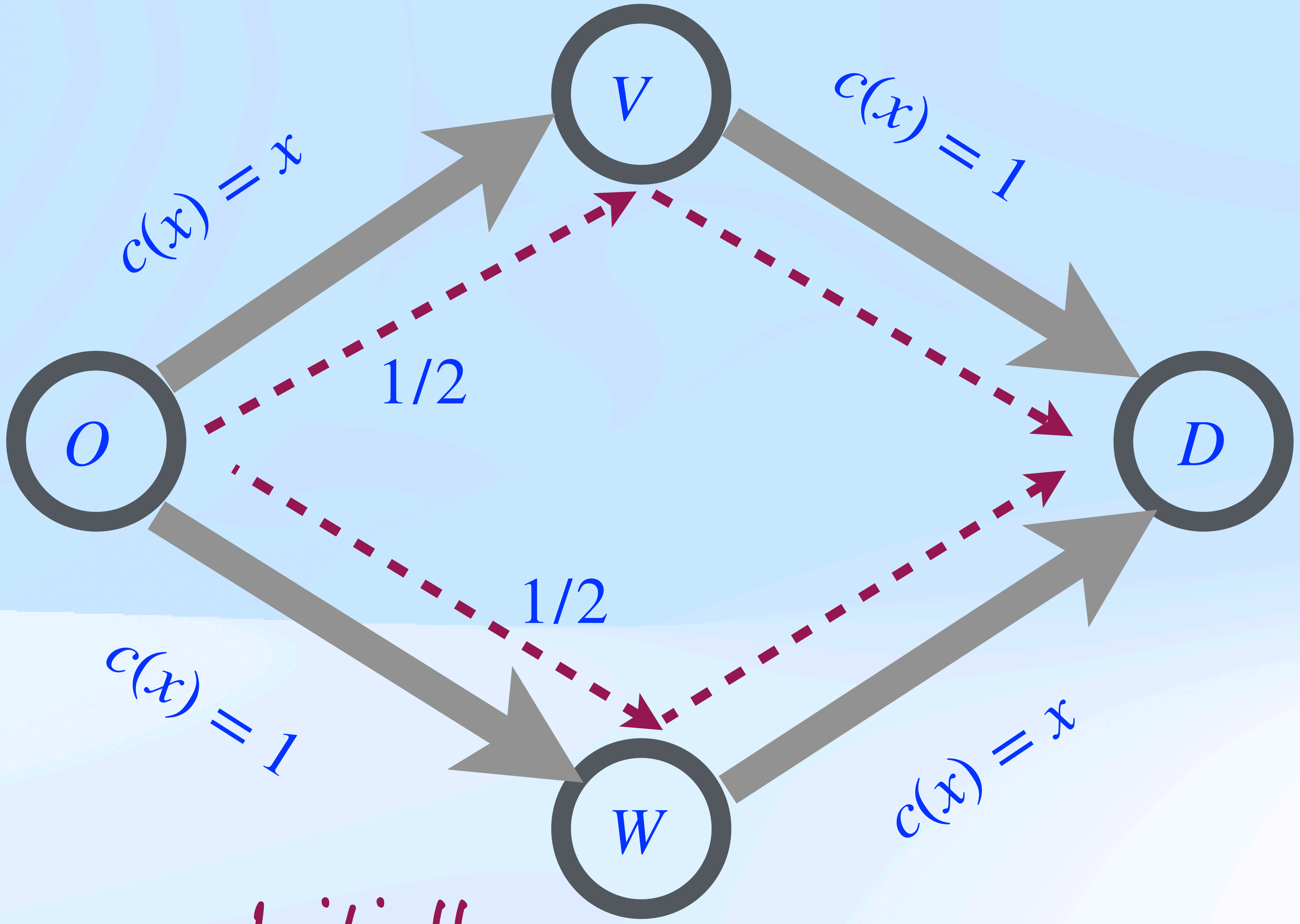


After super-highway

Travel time in DS

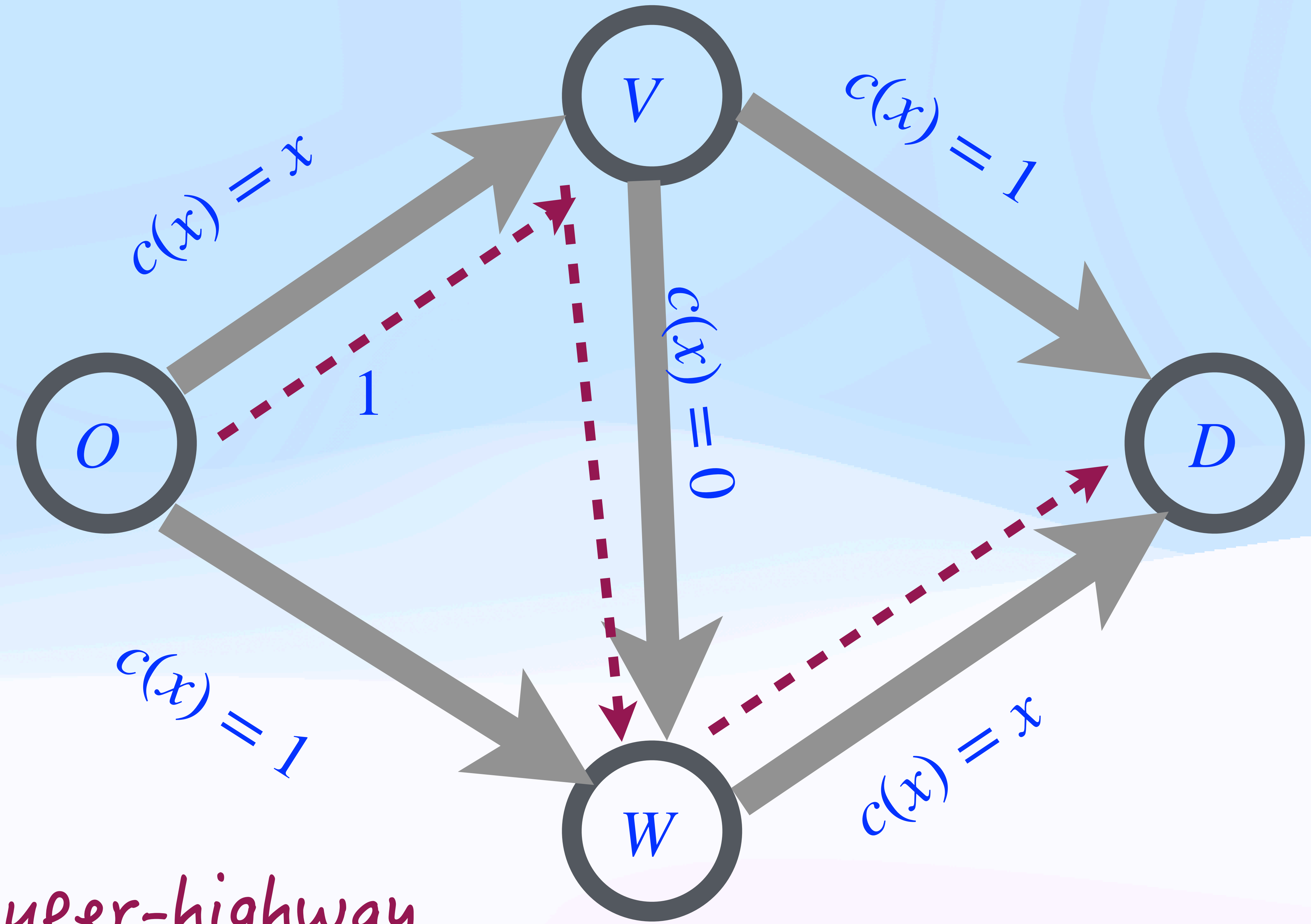
Braess Network, 1965

1 unit of traffic



Initially

Travel time $= 2\left(\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot 1\right) = \frac{3}{2}$

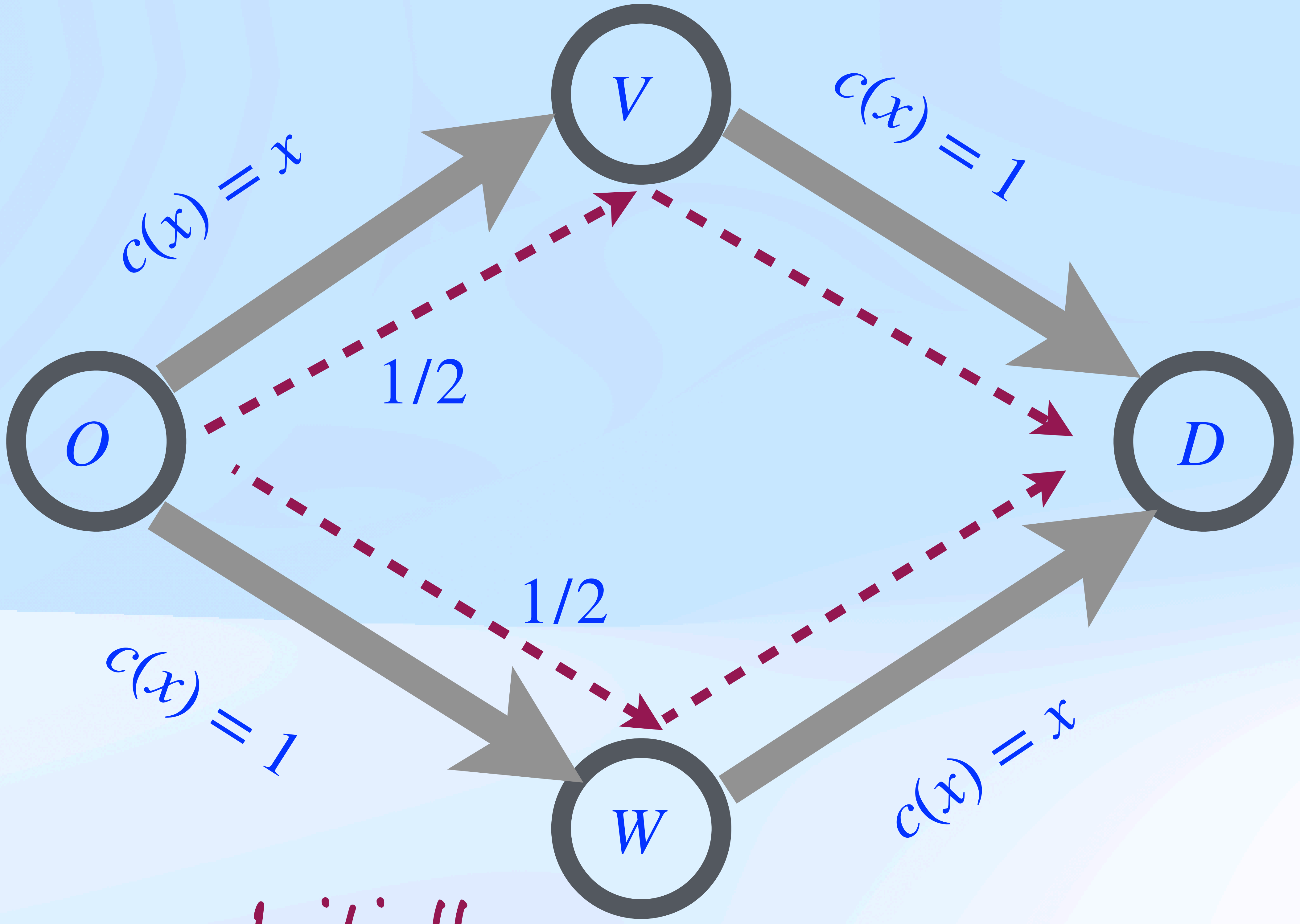


After super-highway

Travel time in DS $= 1\cdot 1 + 0 + 1\cdot 1 = 2$

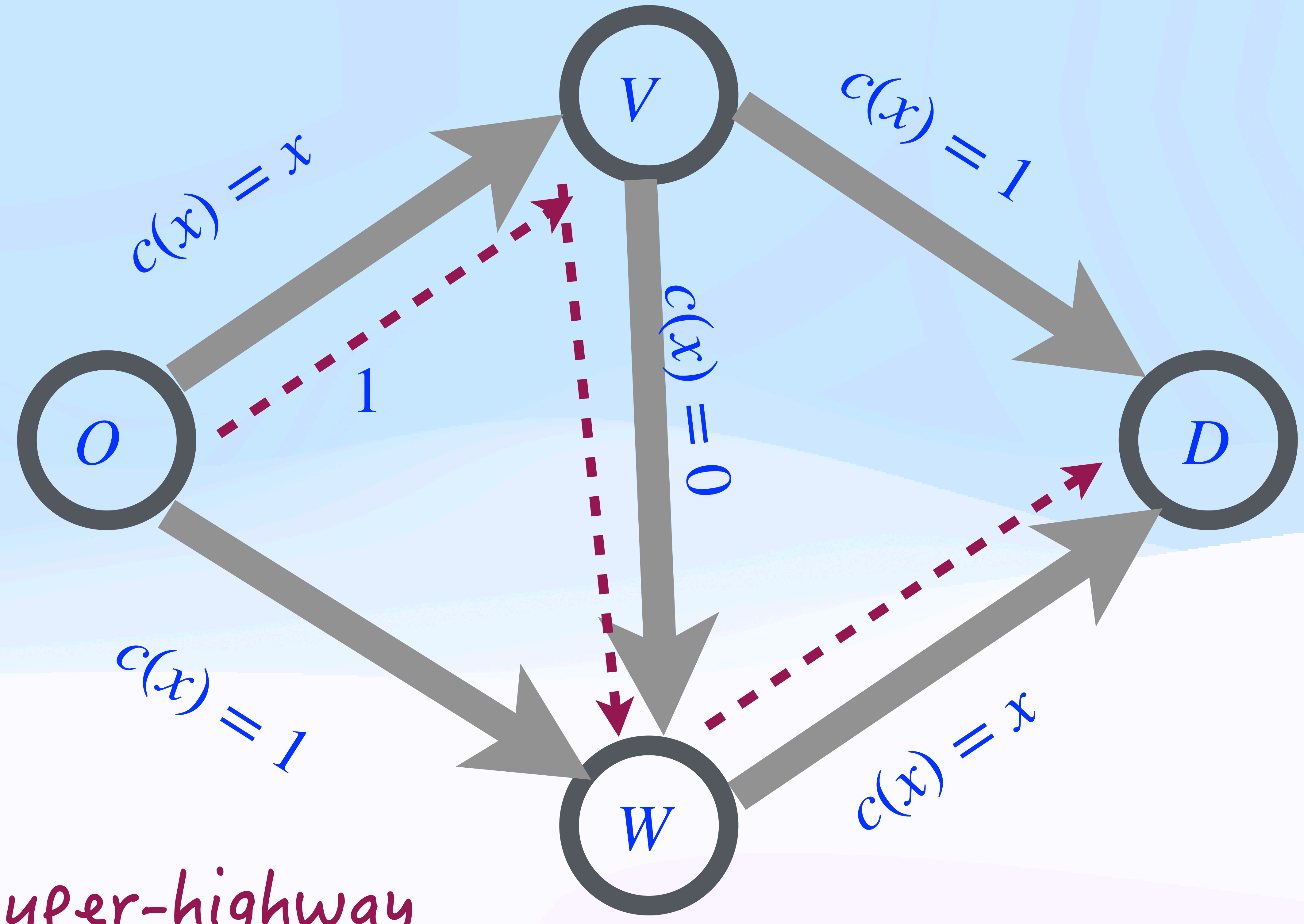
Braess Network, 1965

1 unit of traffic



Initially

Travel time $= 2\left(\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot 1\right) = \frac{3}{2}$



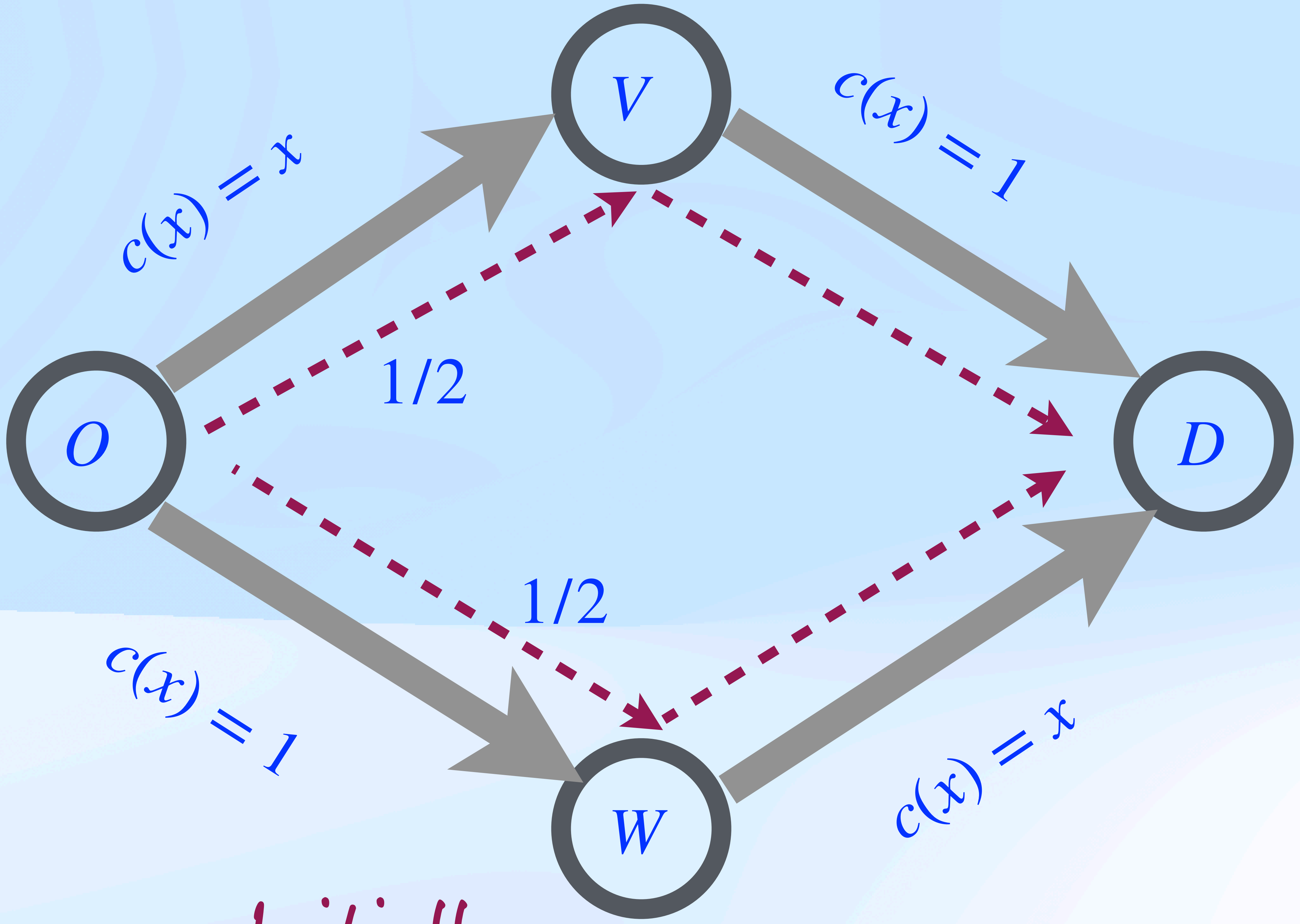
After super-highway

Travel time in DS $= 1\cdot 1 + 0 + 1\cdot 1 = 2$

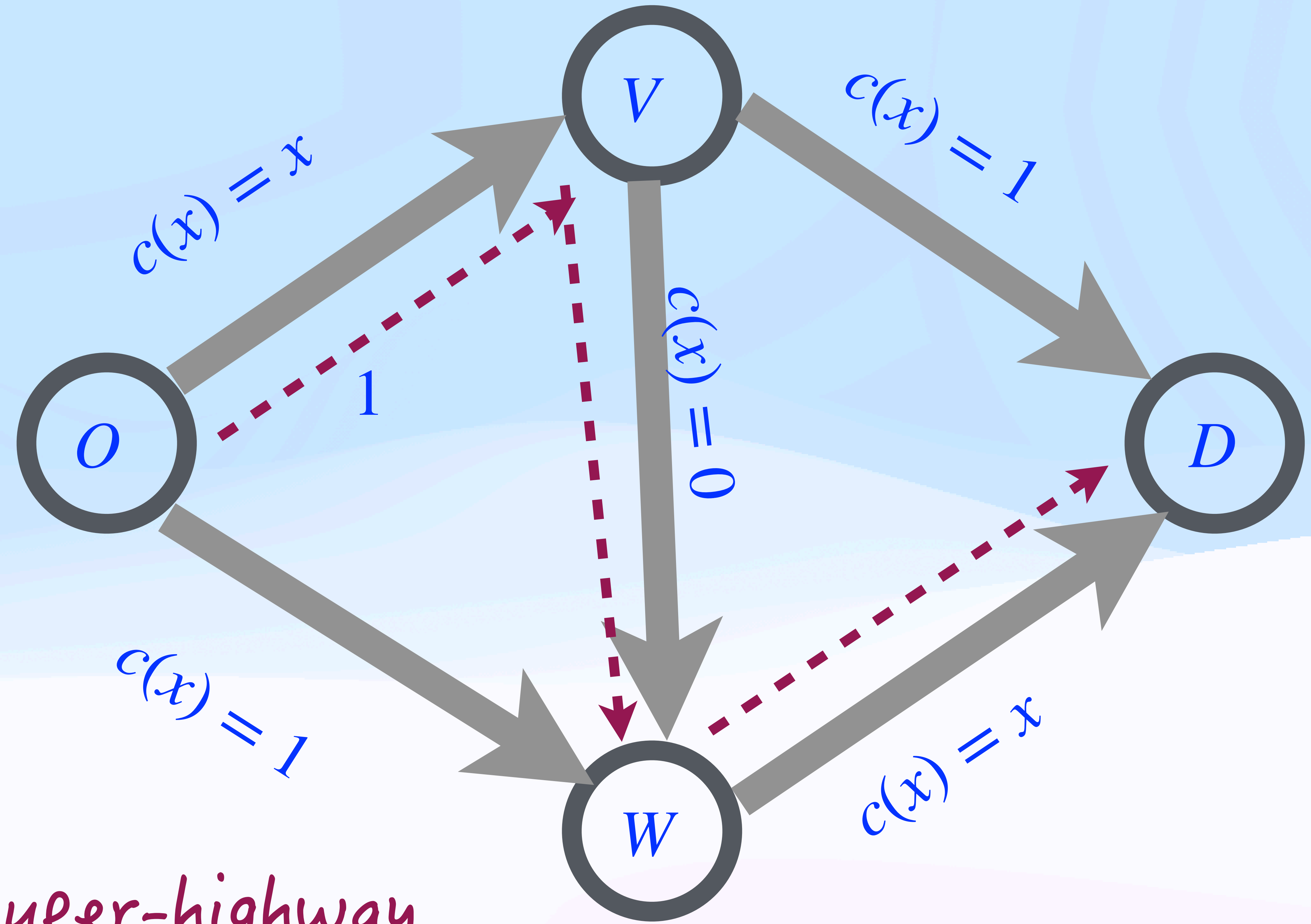
It is taking more time!!

Braess Network, 1965

1 unit of traffic



Initially



After super-highway

Travel time $= 2\left(\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot 1\right) = \frac{3}{2}$

Travel time in DS $= 1\cdot 1 + 0 + 1\cdot 1 = 2$

Price Of Selfish behavior $= \frac{2}{3/2} = \frac{4}{3}$

It is taking more time!!

What is the price of selfish behaviour ?

What is the price of selfish behaviour ?

In <u>Pigou</u> network loss due to selfish behaviour is

33 %

What is the price of selfish behaviour ?

In Pigou network loss due to selfish behaviour is

33 %

What if the network structure is more complex ?

What is the price of selfish behaviour ?

In Pigou network loss due to selfish behaviour is

33 %

What if the network structure is more complex ?

···many more vertices, roads, different source-destination pairs etc

What is the price of selfish behaviour ?

In Pigou network loss due to selfish behaviour is

33 %

What if the network structure is more complex ?

…many more vertices, roads, different source-destination pairs etc

What if the cost functions are more complex ?

What is the price of selfish behaviour ?

In Pigou network loss due to selfish behaviour is
33 %

What if the network structure is more complex ?

…many more vertices, roads, different source-destination pairs etc

What if the cost functions are more complex ?

Can this get worse ?

What if the network is more complicated ??

What if the network is more complicated ??

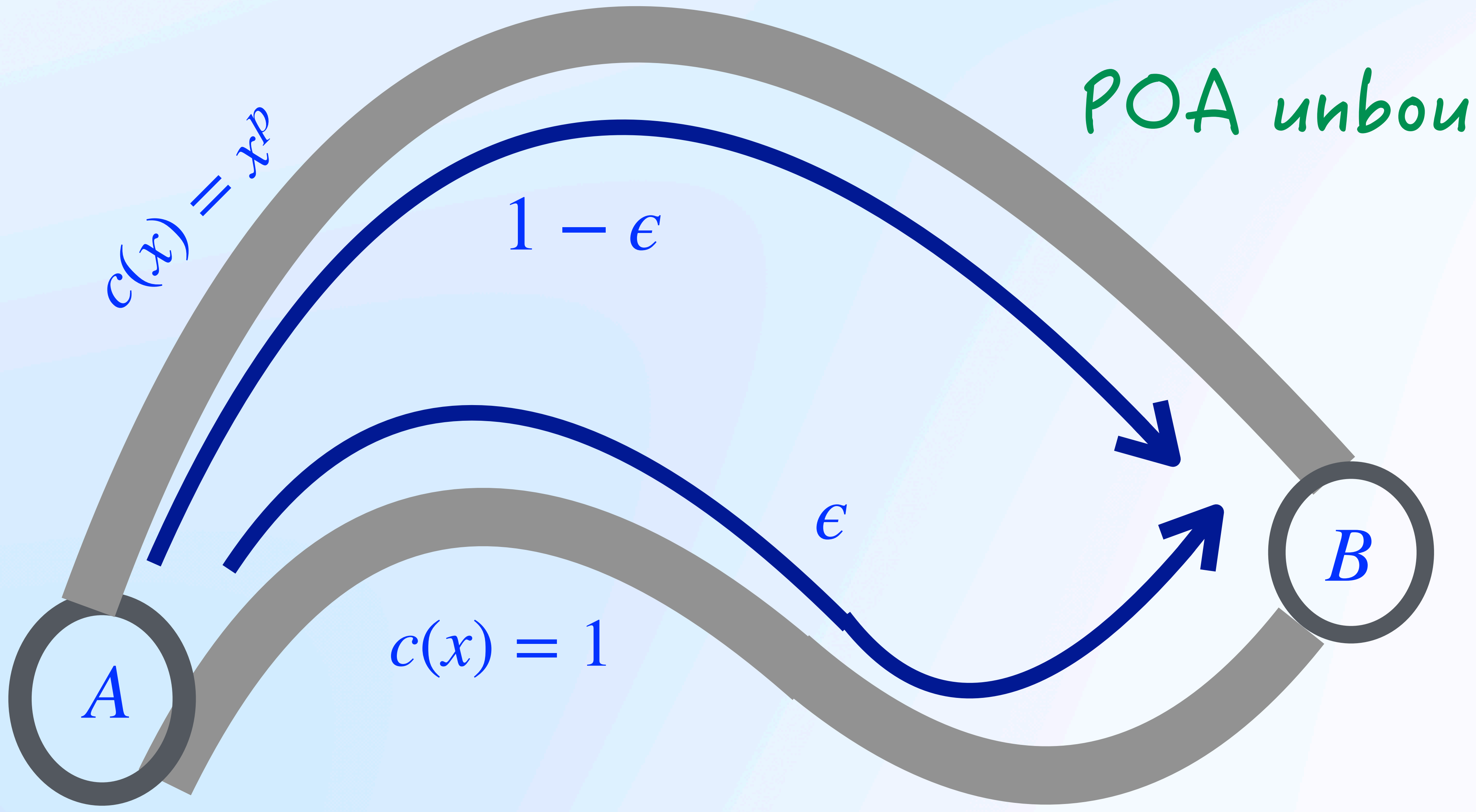
Roughgarden and Tardos 2002

In *every* network routing game with *linear* (or affine) cost function the price of selfishness is $\frac{4}{3}$

What if the network is more complicated ??

Roughgarden and Tardos 2002

In *every* network routing game with *linear* (or affine) cost function the price of selfishness is $\frac{4}{3}$

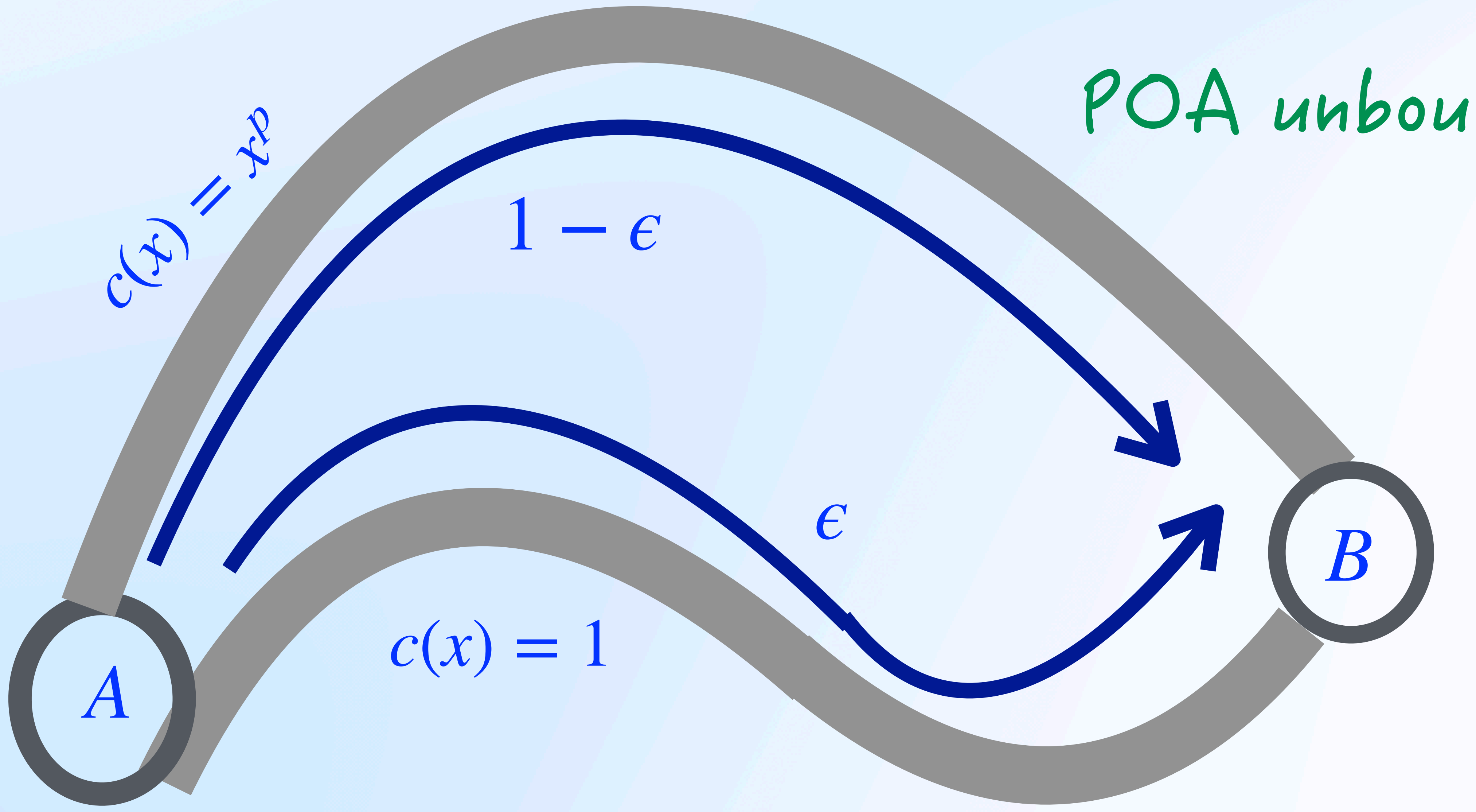


POA unbounded!

What if the network is more complicated ??

Roughgarden and Tardos 2002

In <u>every</u> network routing game with <u>linear</u> (or affine) cost function the price of selfishness is $\frac{4}{3}$



POA unbounded!

Journey of 80 years!

Moral of the story: Culprits are non-linear cost functions!

Different kinds of selfish routing

Different kinds of selfish routing

Nonatomic Selfish Routing

Different kinds of selfish routing

Nonatomic Selfish Routing

Agents have negligible size and individuals have negligible impact on the network

Different kinds of selfish routing

Nonatomic Selfish Routing

Agents have negligible size and individuals have negligible impact on the network

Eg: road traffic, private users of communication network

Different kinds of selfish routing

Nonatomic Selfish Routing

Agents have negligible size and individuals have negligible impact on the network

Eg: road traffic, private users of communication network

Atomic Selfish Routing

Different kinds of selfish routing

Nonatomic Selfish Routing

Agents have negligible size and individuals have negligible impact on the network

Eg: road traffic, private users of communication network

Atomic Selfish Routing

Each agent controls a significant fraction of the overall traffic.

Different kinds of selfish routing

Nonatomic Selfish Routing

Agents have negligible size and individuals have negligible impact on the network

Eg: road traffic, private users of communication network

Atomic Selfish Routing

Each agent controls a significant fraction of the overall traffic.

Eg: an agent could represent an ISP responsible for routing the data of a large number of end users

Some cool facts about congestion games

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Later, Monderer and Shapley proved the converse:

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Later, Monderer and Shapley proved the converse:

Any game with an exact potential function is equivalent to some congestion game.

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Later, Monderer and Shapley proved the converse:

Any game with an exact potential function is equivalent to some congestion game.

In recent times, people have asked:

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Later, Monderer and Shapley proved the converse:

Any game with an exact potential function is equivalent to some congestion game.

In recent times, people have asked: Do potential functions (and thus PNE) exist for more general congestion games ?

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Later, Monderer and Shapley proved the converse:

Any game with an exact potential function is equivalent to some congestion game.

In recent times, people have asked:

Do potential functions (and thus PNE) exist for more general congestion games ?

What is the rate of convergence ?

Some cool facts about congestion games

Research initiated by economist Robert W. Rosenthal in 1973

He proved that every congestion game has a pure strategy Nash equilibrium (PNE)

…not a common phenomenon!

In fact, he proved that every congestion game is an exact potential game.

Later, Monderer and Shapley proved the converse:

Any game with an exact potential function is equivalent to some congestion game.

In recent times, people have asked:

Do potential functions (and thus PNE) exist for more general congestion games ?

What is the rate of convergence ?

What is the computational complexity of finding an equilibrium?

Thank You!

Proof Details

We will prove

Among all networks with cost function $\mathscr{C}$, the largest POA is achieved in a Pigou-like network.

More formally

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the $POA = \dfrac{\text{Flow in DSE}}{\text{Optimum flow}}$ is at most $\alpha(\mathscr{C})$, where

More formally

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the

$POA = \dfrac{\text{Flow in DSE}}{\text{Optimum flow}}$ is at most $\alpha(\mathscr{C})$, where

$$\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

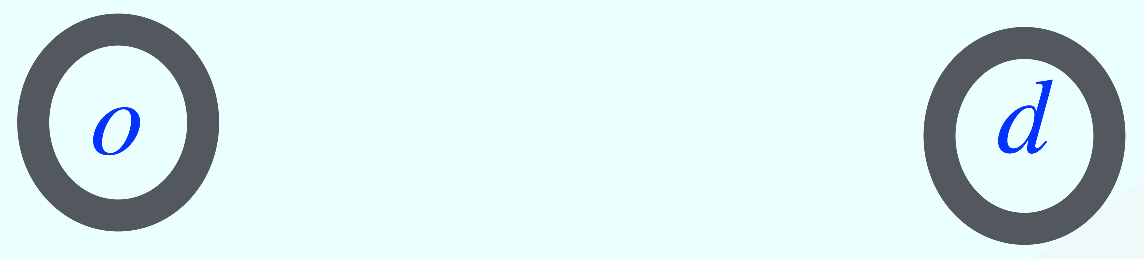
Let's start from the start …

Let's start from the start ⋯

1. Two vertices, origin o and destination d

Let's start from the start …

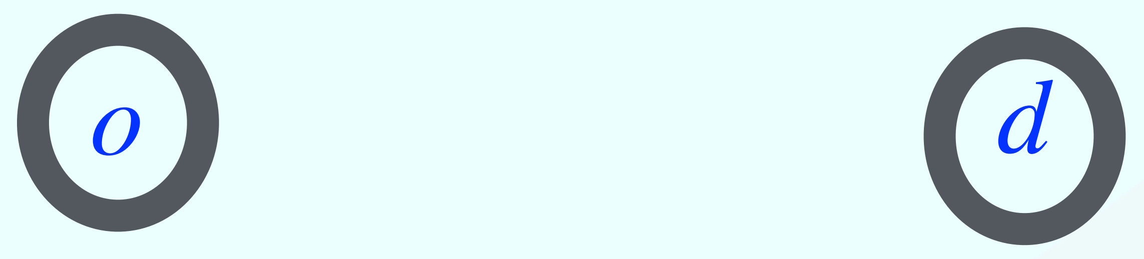
1. Two vertices, origin o and destination d



Let's start from the start ···

1. Two vertices, origin o and destination d

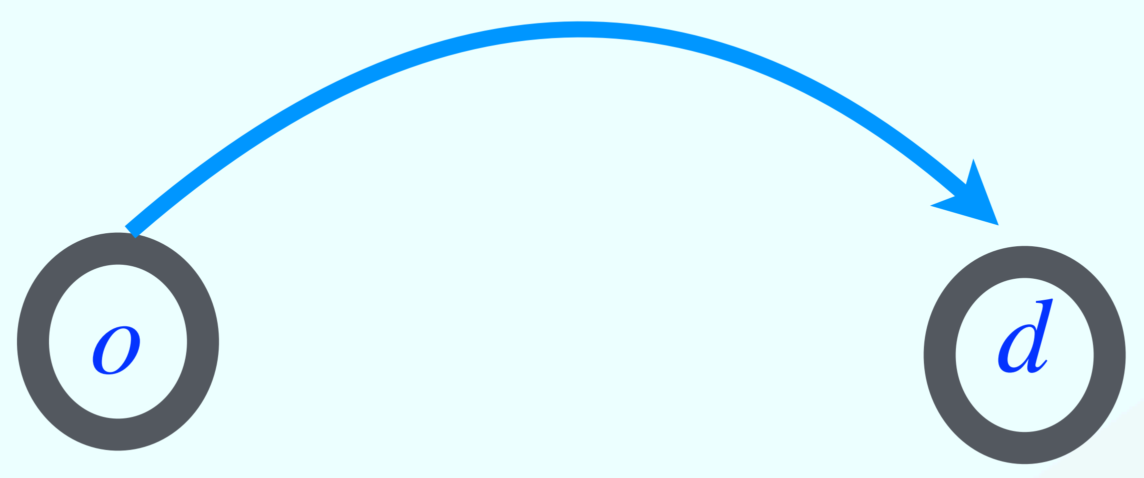
2. Two edges from o to d, and upper and lower edge



Let's start from the start …

1. Two vertices, origin o and destination d

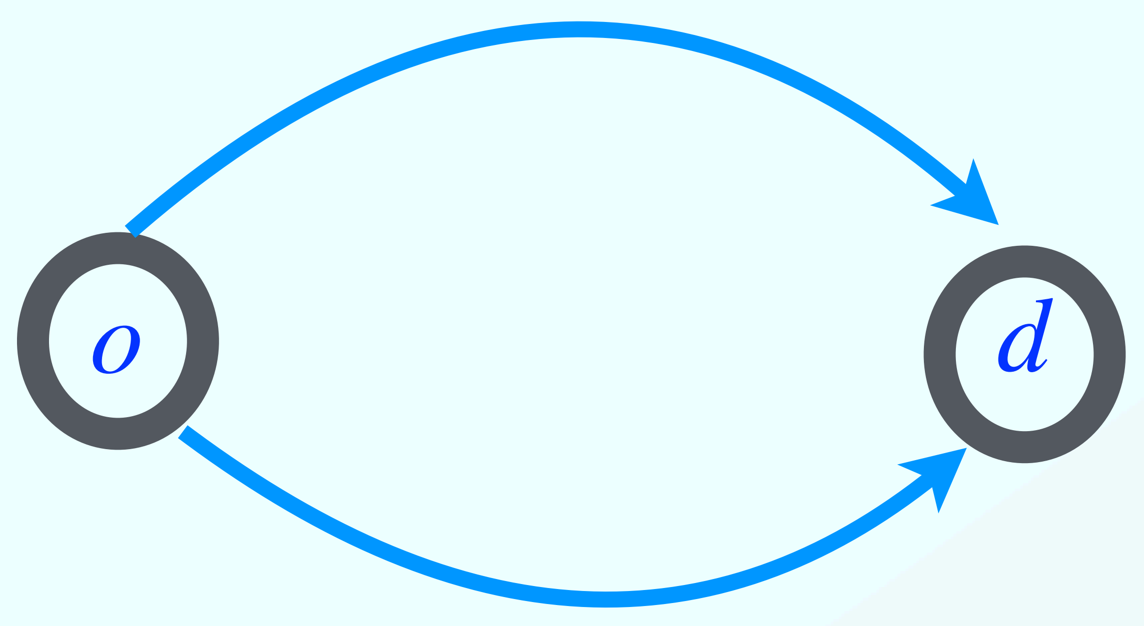
2. Two edges from o to d, and upper and lower edge



Let's start from the start ...

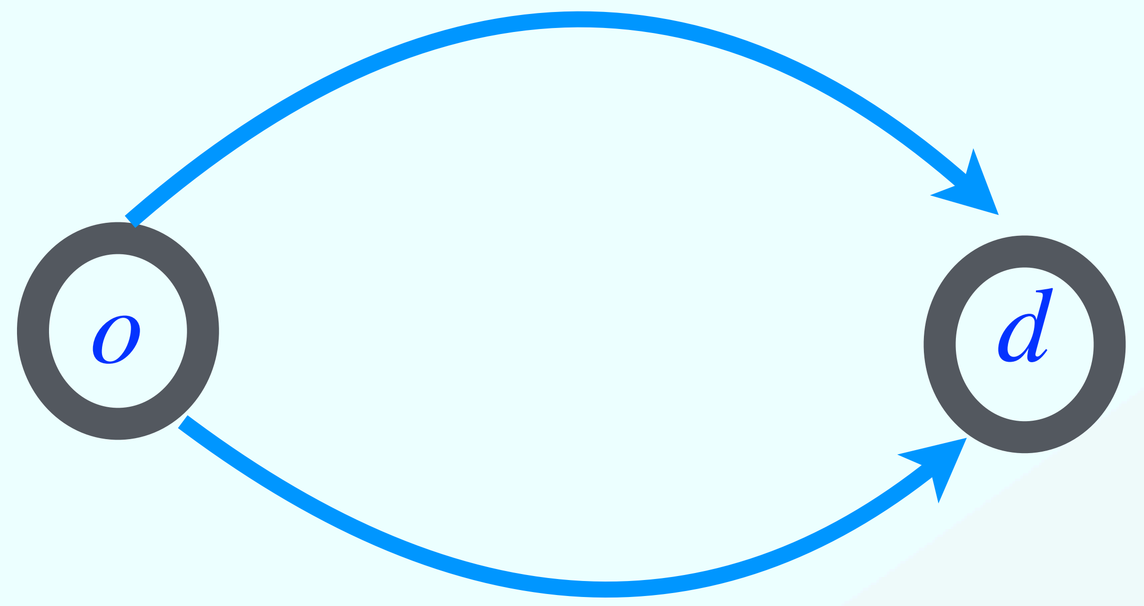
1. Two vertices, origin o and destination d

2. Two edges from o to d, and upper and lower edge



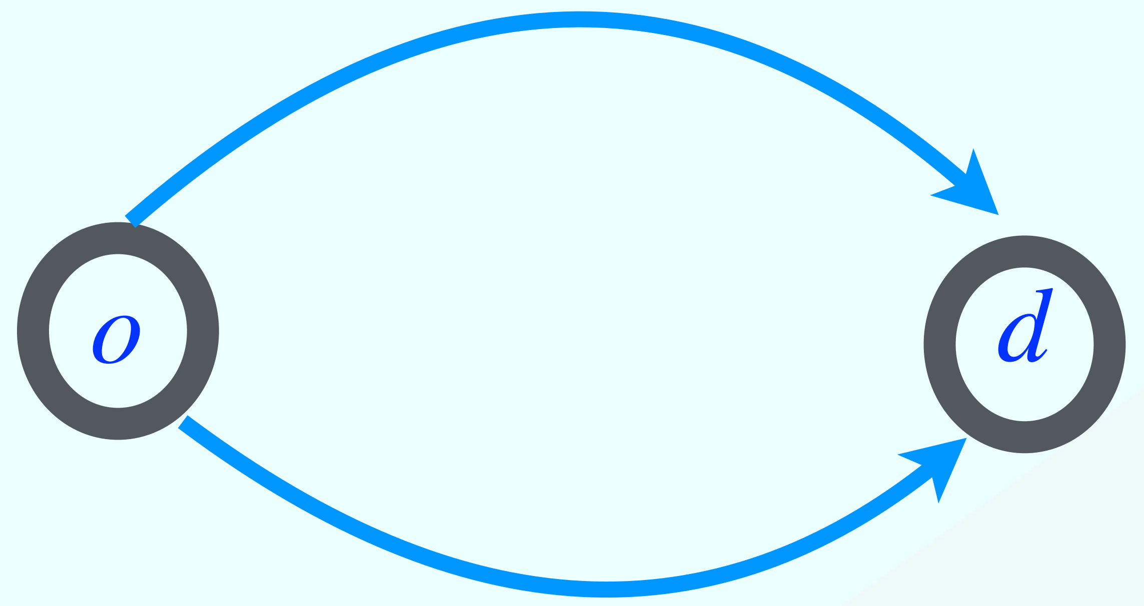
Let's start from the start …

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r



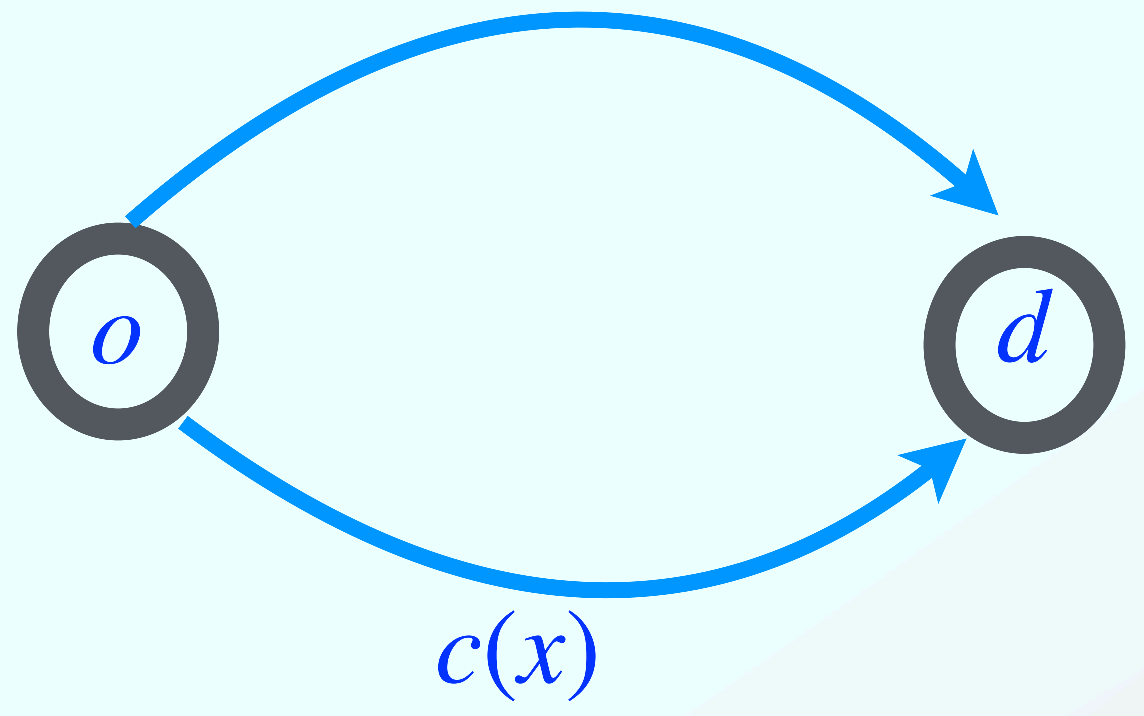
Let's start from the start ···

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.

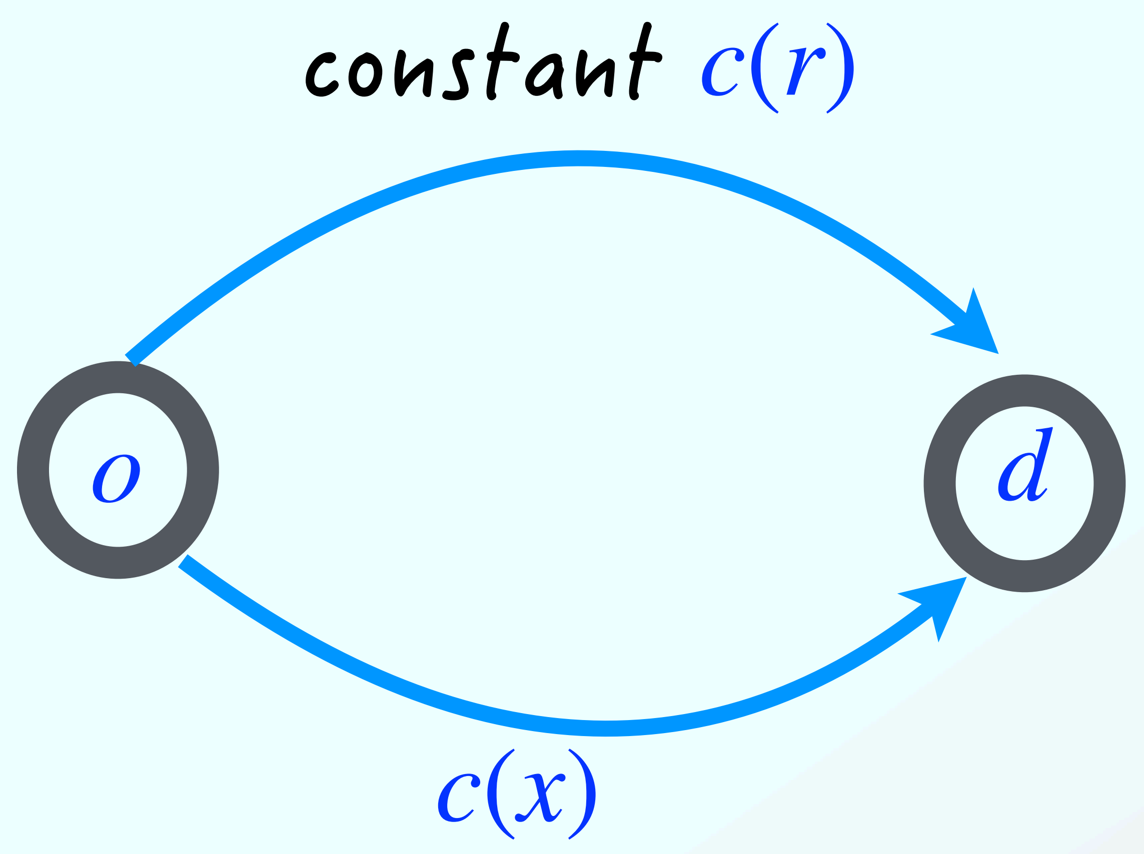


Let's start from the start ···

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.



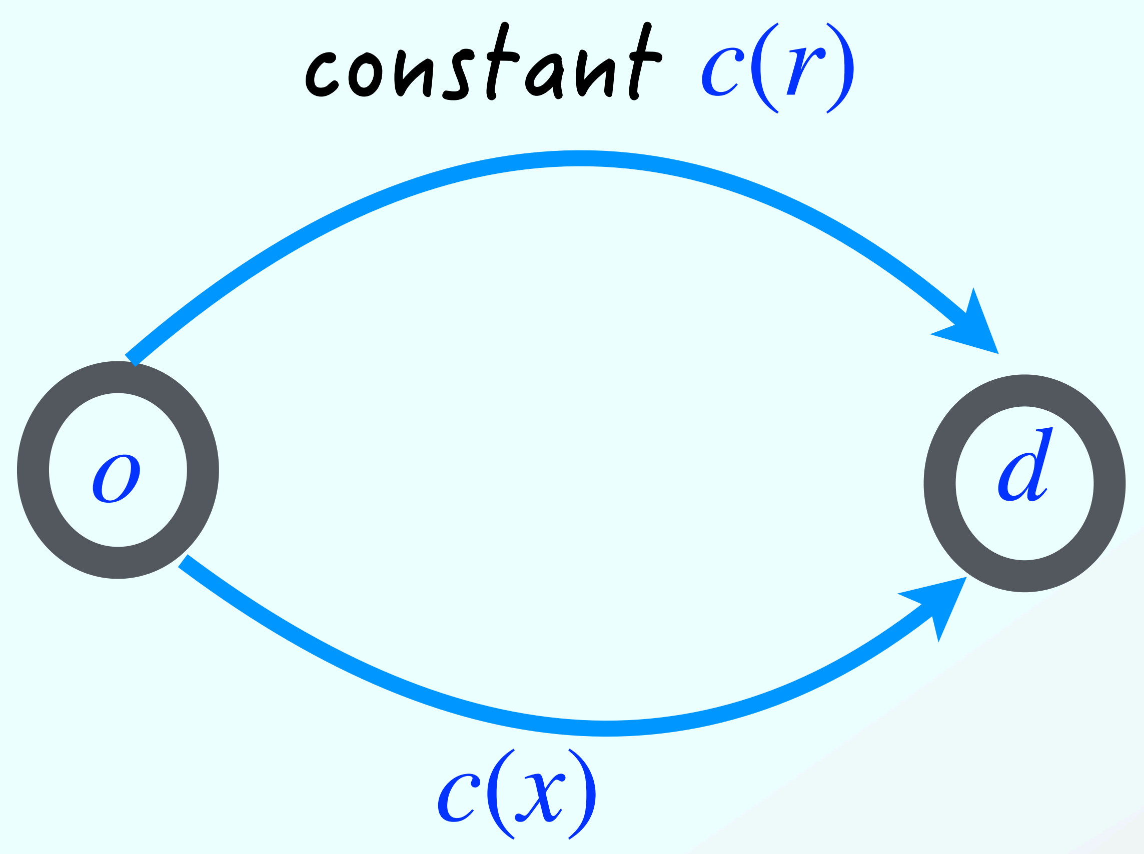
Let's start from the start ⋯



1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.

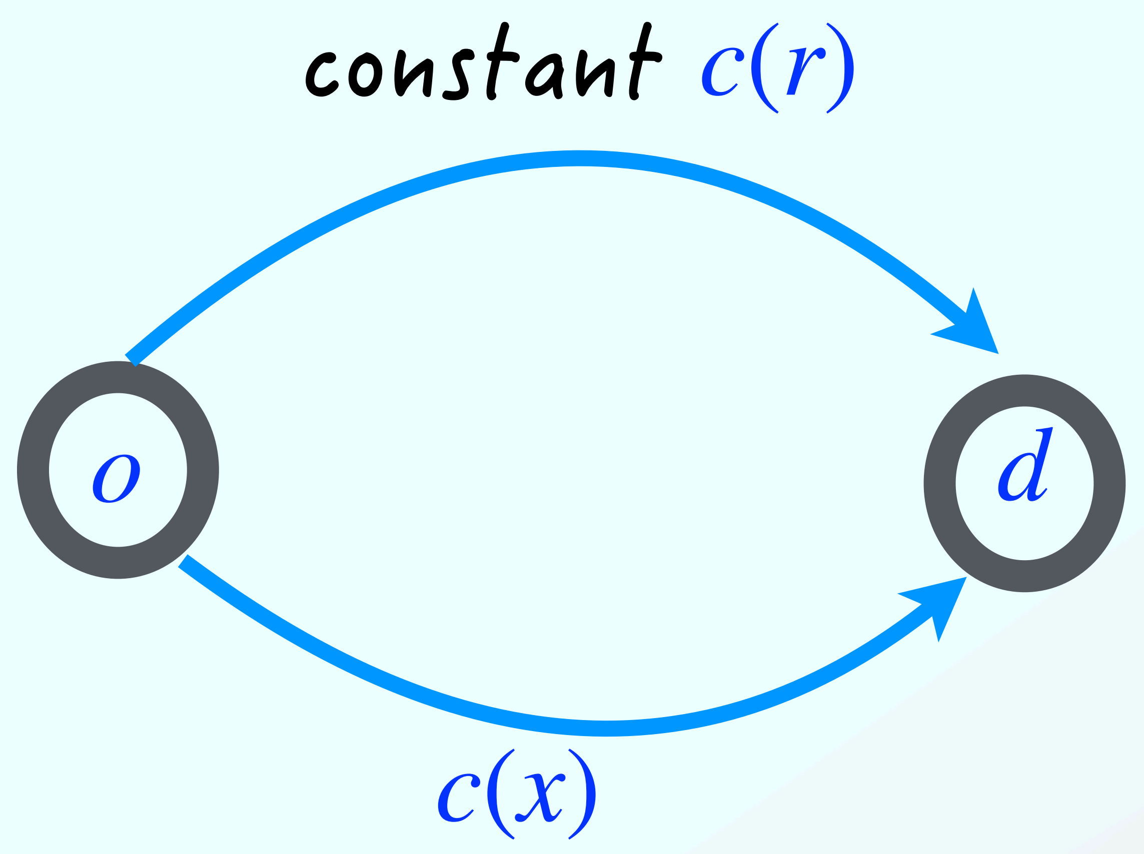
Let's start from the start …

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.



Let's start from the start ⋯

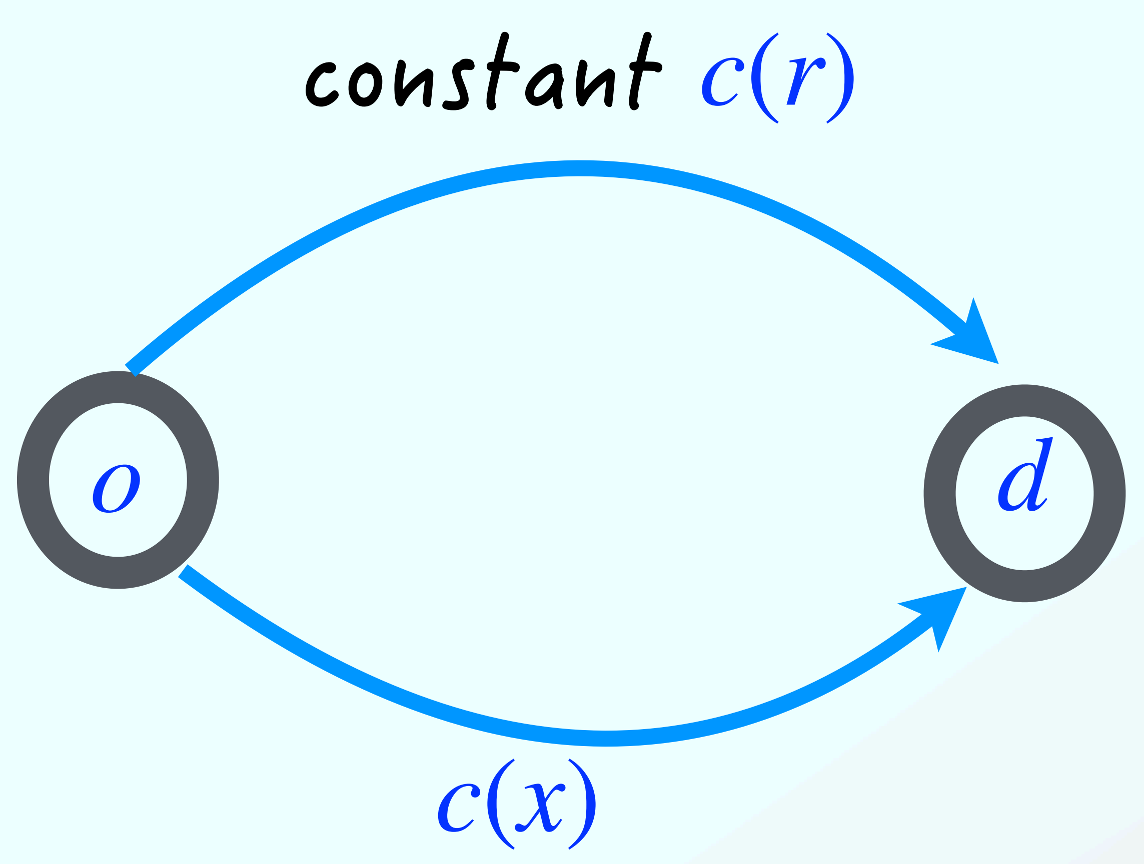
1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.



DS is to send flow via lower edge

Let's start from the start …

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.

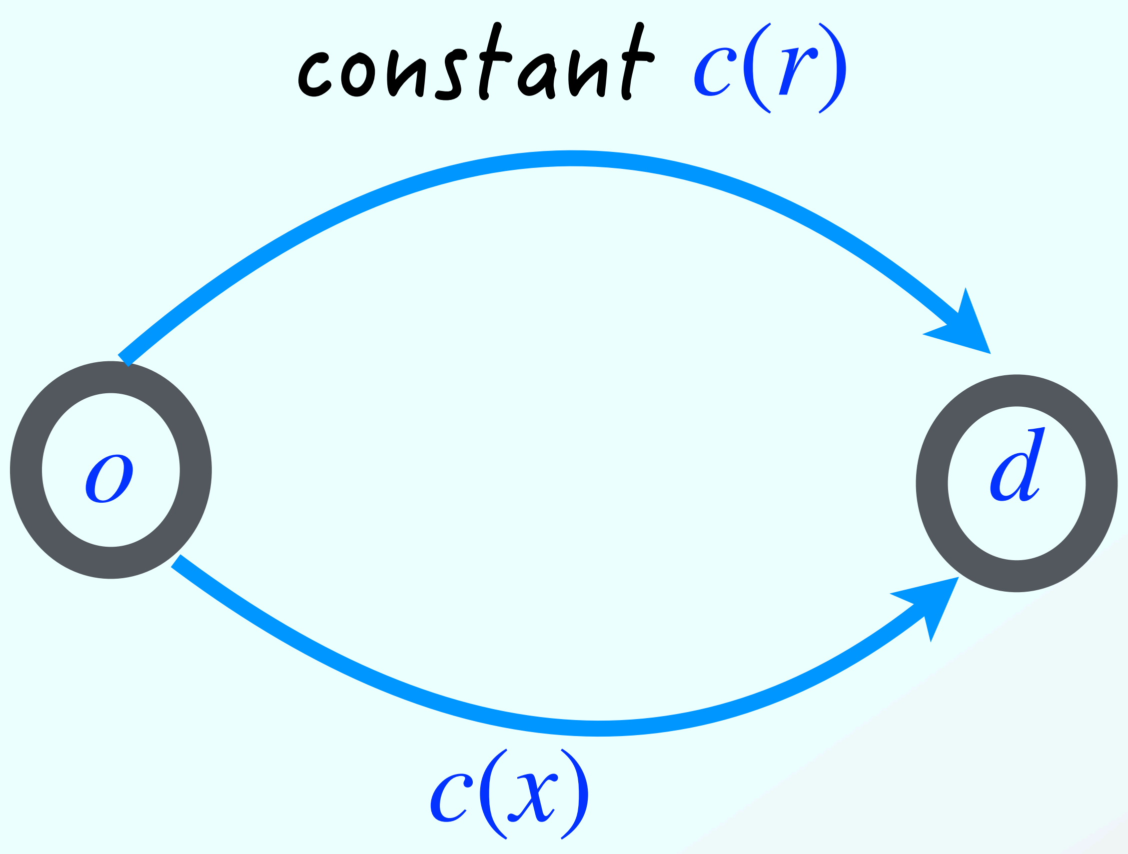


DS is to send flow via lower edge

…no less attractive than the constant cost c(r), even when fully congested!

Let's start from the start ···

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.



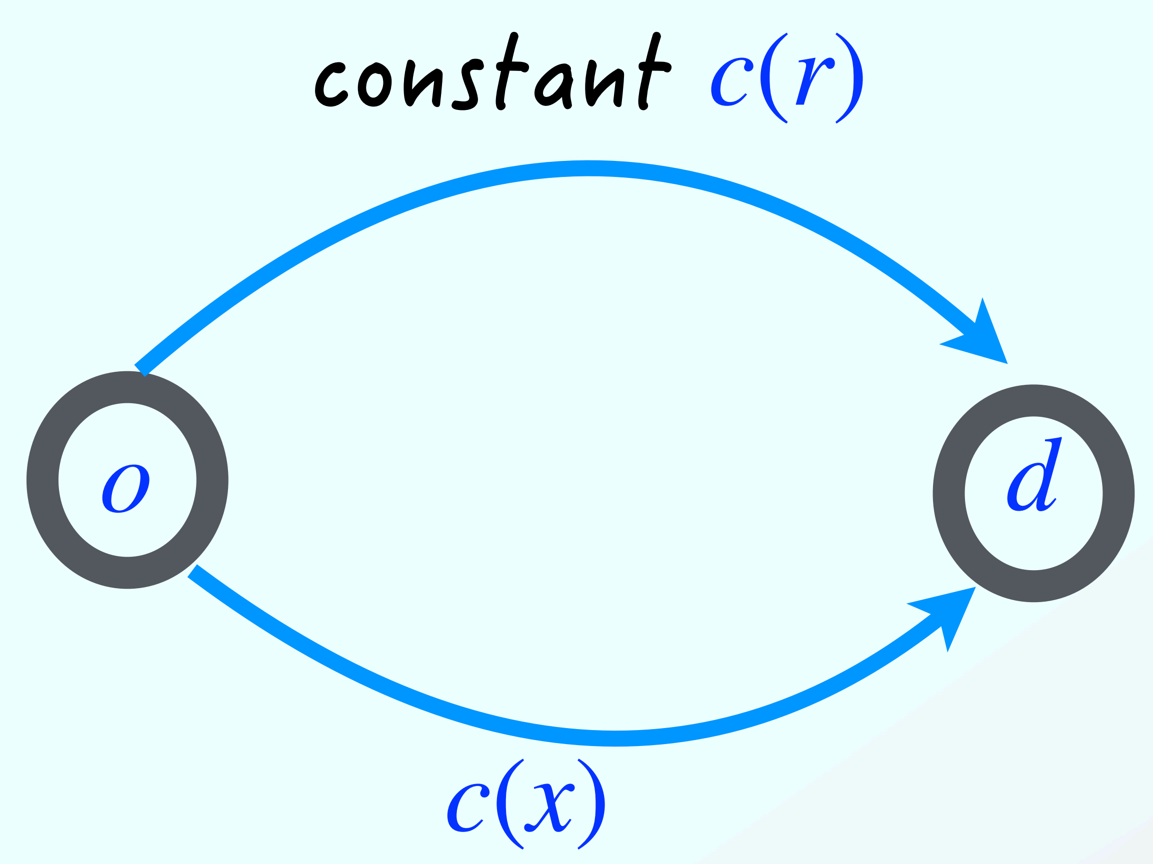
DS is to send flow via lower edge

Travel time in DSE $= r \cdot c(r)$

···no less attractive than the constant cost c(r), even when fully congested!

Let's start from the start ⋯

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.



DS is to send flow via lower edge

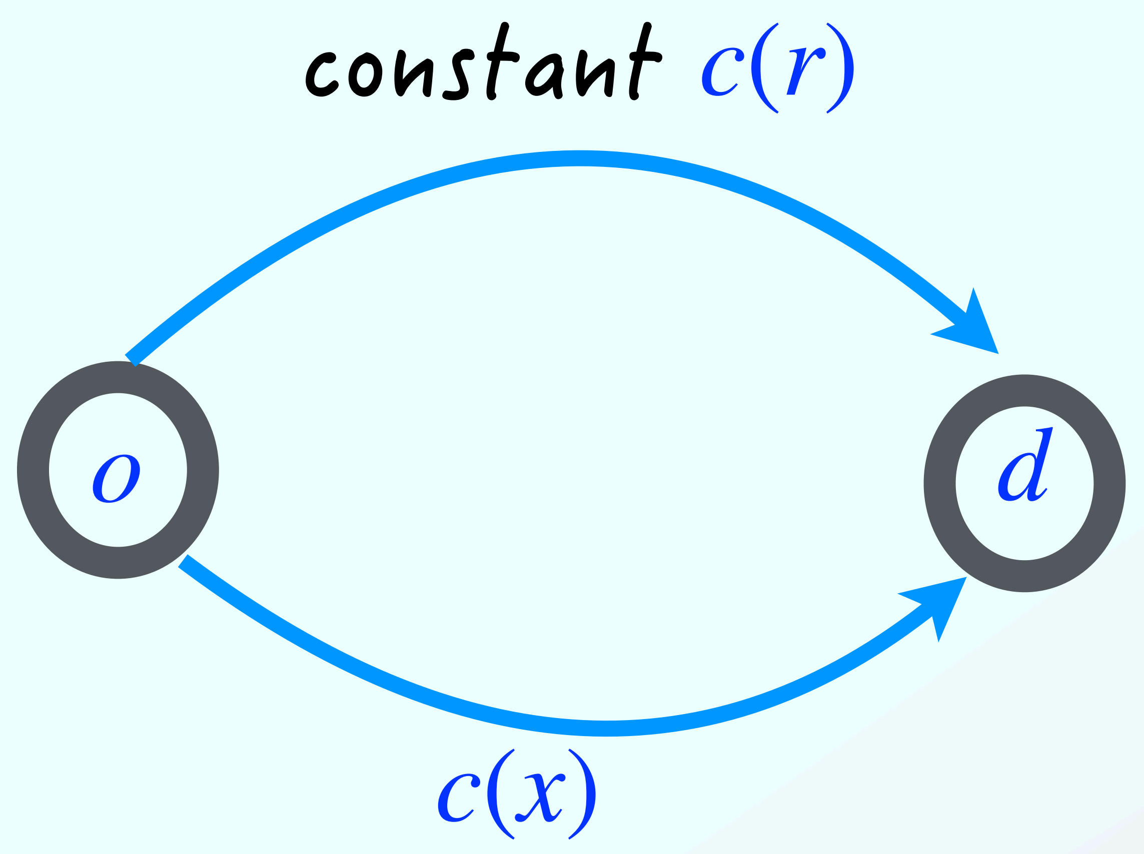
Travel time in DSE $= r \cdot c(r)$

⋯no less attractive than the constant cost c(r), even when fully congested!

Min possible total travel time is $\inf_{0 \le x \le r} \{x \cdot c(x) + (r - x) \cdot c(r)\}$

Let's start from the start ⋯

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.



DS is to send flow via lower edge

Travel time in DSE $= r \cdot c(r)$

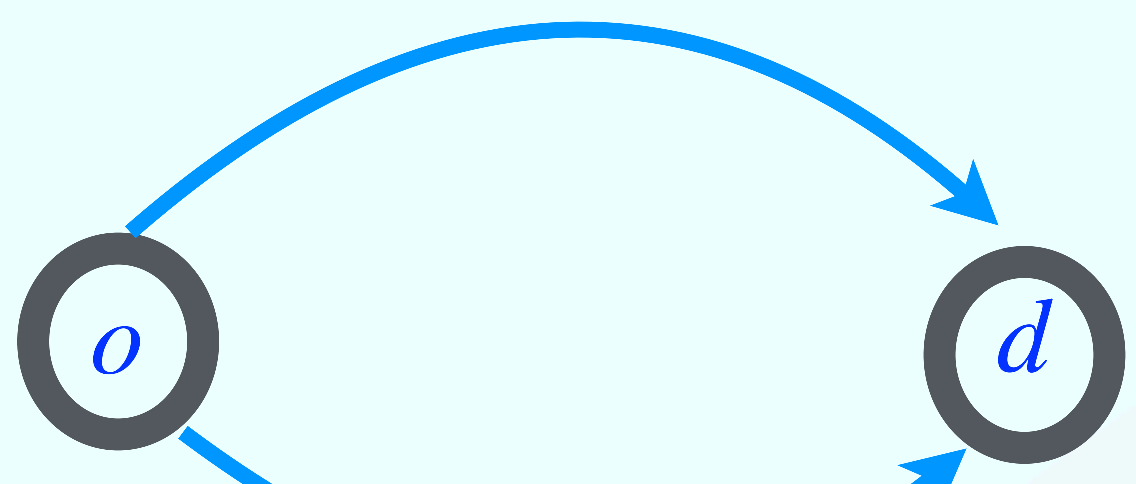
⋯no less attractive than the constant cost c(r), even when fully congested!

Min possible total travel time is $\inf_{0 \le x \le r} \{x \cdot c(x) + (r - x) \cdot c(r)\}$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}}$$

Let's start from the start …

1. Two vertices, origin o and destination d
2. Two edges from o to d, and upper and lower edge
3. A non-negative traffic rate r
4. A cost function $c(x)$ in the lower edge.
5. The (constant) cost function $c(r)$ on the upper edge.



DS is to send flow via lower edge

Travel time in DSE $= r \cdot c(r)$

…no less attractive than the constant cost c(r), even when fully congested!

Min possible total travel time is $\inf_{0 \le x \le r} \{x \cdot c(x) + (r - x) \cdot c(r)\}$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}} = \sup_{x \ge 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}}$$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}} = \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}} = \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

Let $\mathscr{C}$ be an arbitrary set of non-negative, continuous, and nondecreasing cost functions.

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}} = \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

Let $\mathscr{C}$ be an arbitrary set of non-negative, continuous, and nondecreasing cost functions.

Pigou bound $\alpha(\mathscr{C})$ is the largest POA in a Pigou-like network in which the lower edge cost function belongs to $\mathscr{C}$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}} = \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

Let $\mathscr{C}$ be an arbitrary set of non-negative, continuous, and nondecreasing cost functions.

Pigou bound $\alpha(\mathscr{C})$ is the largest POA in a Pigou-like network in which the lower edge cost function belongs to $\mathscr{C}$

$$\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \left\{ \sup_{r \geq 0} \left\{ \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\} \right\} \right\}$$

$$\text{POA} = \frac{\text{Travel time in DSE}}{\text{Min possible travel time}} = \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$$

Let $\mathscr{C}$ be an arbitrary set of non-negative, continuous, and nondecreasing cost functions.

Pigou bound $\alpha(\mathscr{C})$ is the largest POA in a Pigou-like network in which the lower edge cost function belongs to $\mathscr{C}$

$$\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \left\{ \sup_{r \geq 0} \left\{ \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\} \right\} \right\}$$

Worst POA with polynomial cost functions with positive coefficient

Description	Typical Representative	Price of Anarchy
Linear	$ax + b$	$4/3$
Quadratic	$ax^2 + bx + c$	$\frac{3\sqrt{3}}{3\sqrt{3}-2} \approx 1.6$
Cubic	$ax^3 + bx^2 + cx + d$	$\frac{4\sqrt[3]{4}}{4\sqrt[3]{4}-3} \approx 1.9$
Quartic	$ax^4 + bx^3 + cx^2 + dx + e$	$\frac{5\sqrt[4]{5}}{5\sqrt[4]{5}-4} \approx 2.2$
Degree $\leq p$	$\sum_{i=0}^{p} a_i x^i$	$\frac{(p+1)\sqrt[p]{p+1}}{(p+1)\sqrt[p]{p+1}-p} \approx \frac{p}{\ln p}$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the *POA* is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

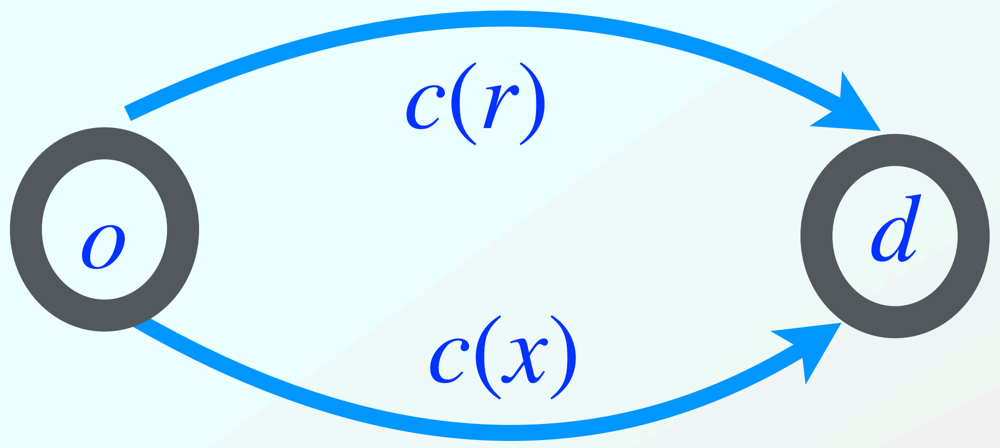
Proof sketch:

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the *POA* is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.

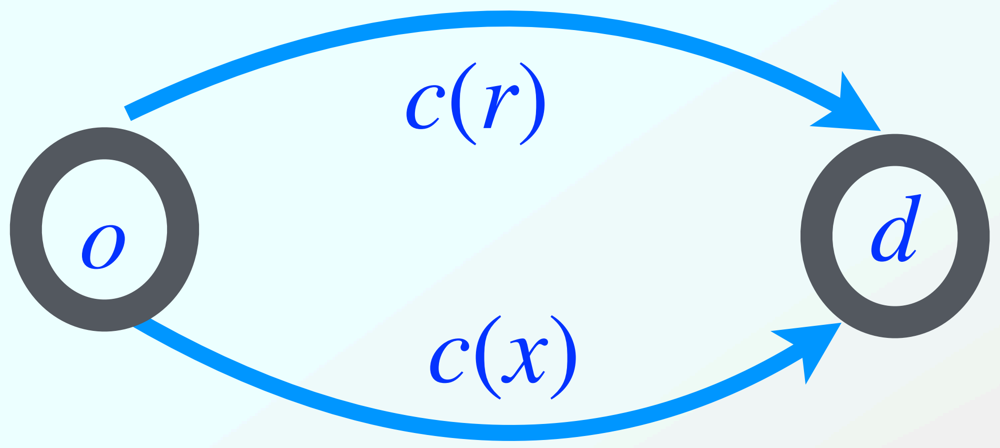
For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.



For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x) \cdot c(r)} \right\}$

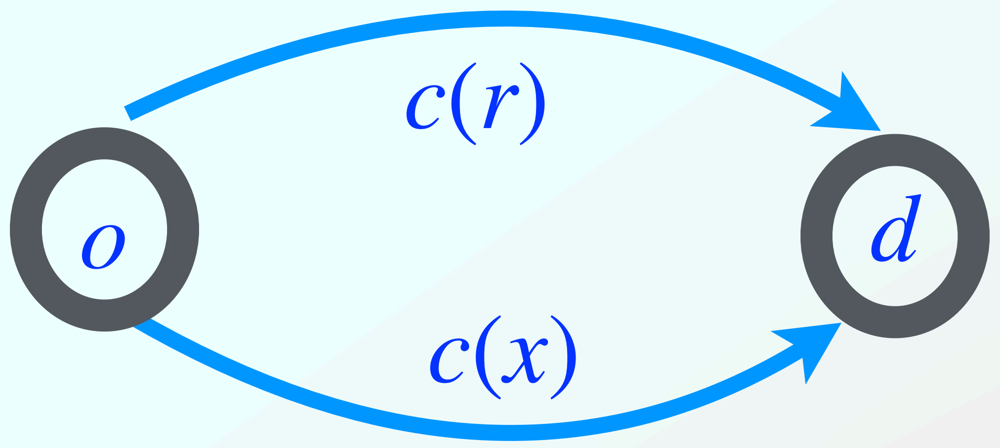
Proof sketch: Let $G=(V,E)$ be a SRN with r unit of traffic between o to d.



Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the *POA* is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.

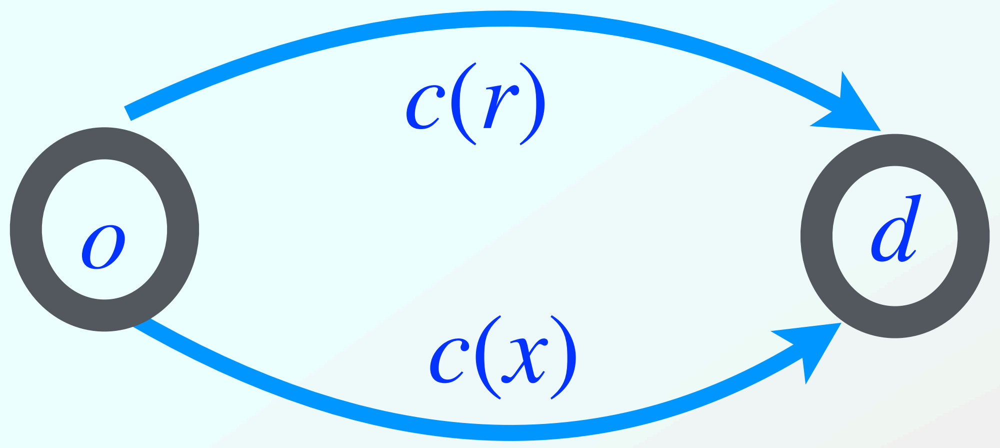


Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.

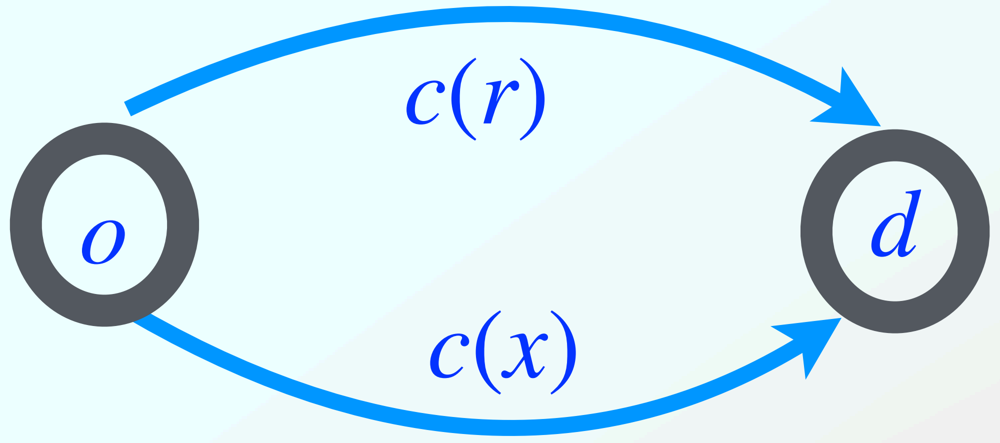


Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths Non-negative vector $\{f_P\}_{P \in \mathscr{P}}$ with $\sum_{p \in \mathscr{P}} f_P = r$

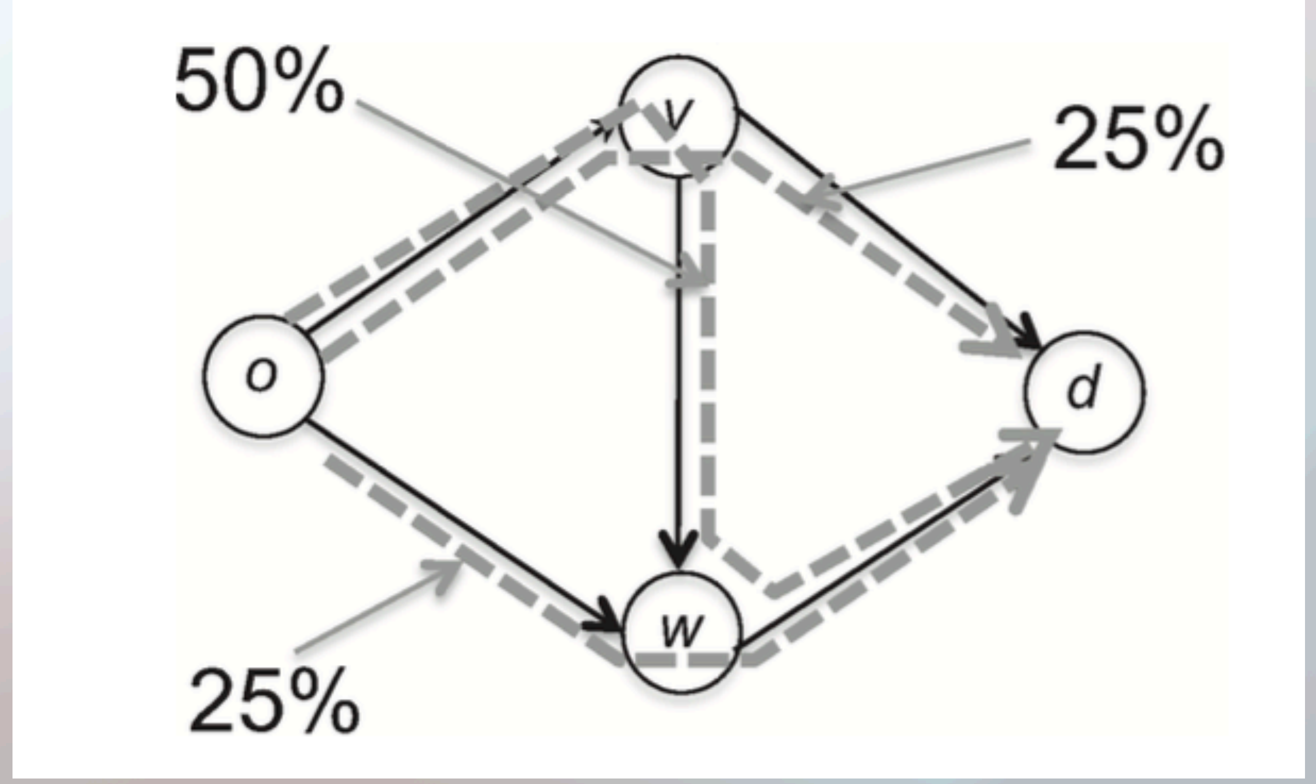
For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.



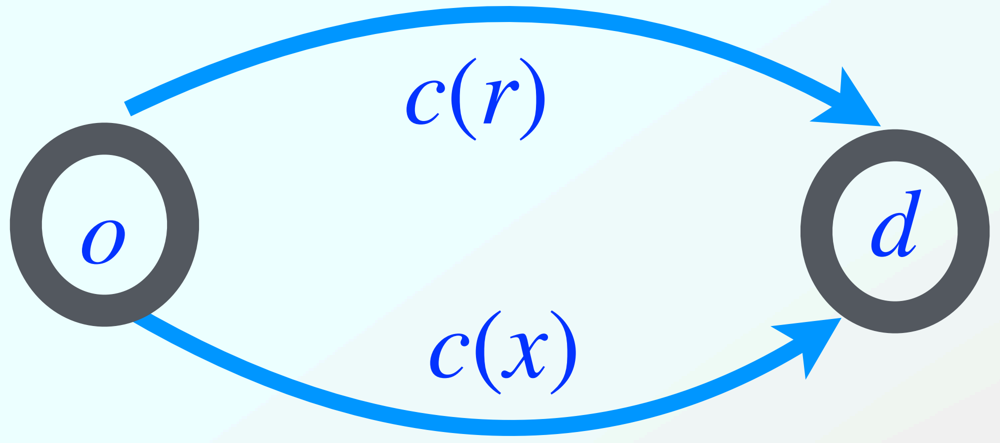
Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths Non-negative vector $\{f_P\}_{P \in \mathscr{P}}$ with $\sum_{p \in \mathscr{P}} f_P = r$



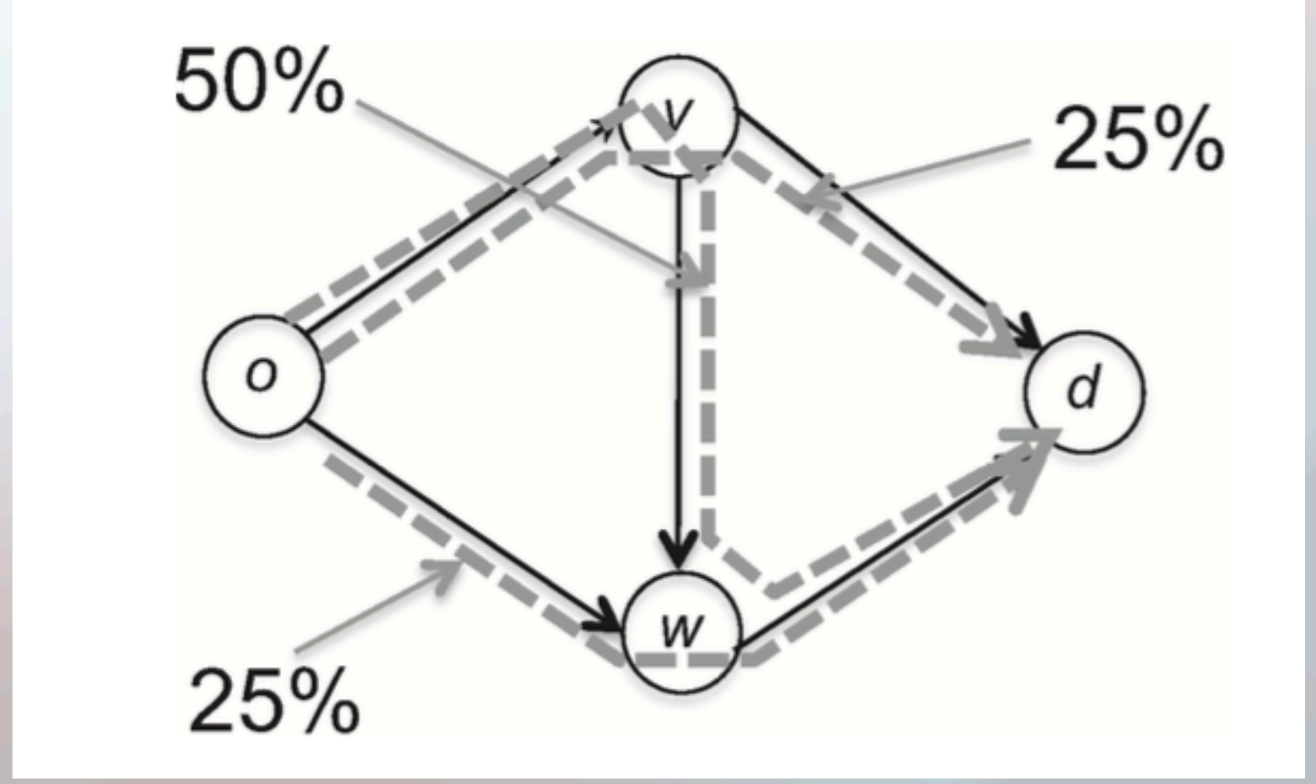
For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x)\cdot c(r)} \right\}$

Proof sketch: Let $G=(V,E)$ be a SRN with r unit of traffic between o to d.



Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o-d$ paths Non-negative vector $\{f_P\}_{P\in\mathscr{P}}$ with $\sum_{p\in\mathscr{P}} f_P = r$

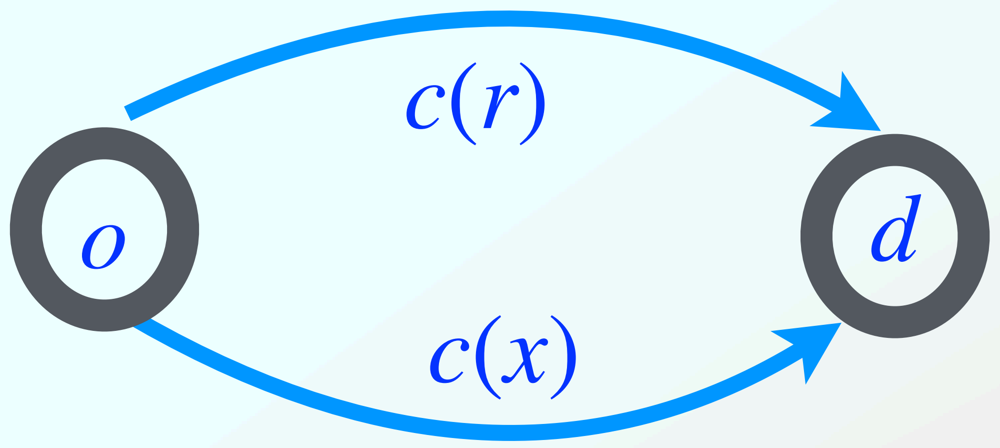


$f_{(o,v)} = f_{(w,d)} = \frac{3}{4}$ $f_{(ow)} = f_{(vd)} = \frac{1}{4}$

$f_{(vw)} = \frac{1}{2}$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

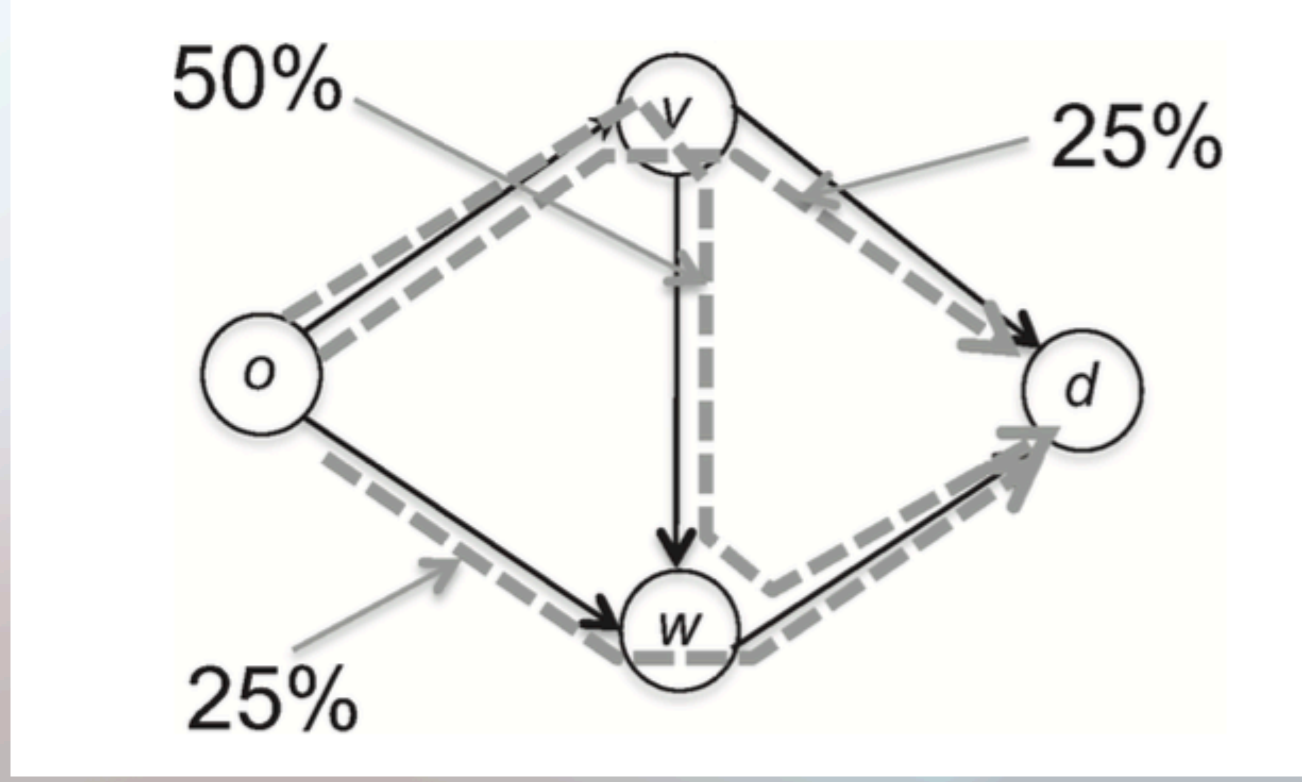
Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.



Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths Non-negative vector $\{f_P\}_{P \in \mathscr{P}}$ with $\sum_{p \in \mathscr{P}} f_P = r$

For an edge $e \in E$ and flow f we write $f_e = \sum_{P \in \mathscr{P}: e \in P} f_P$

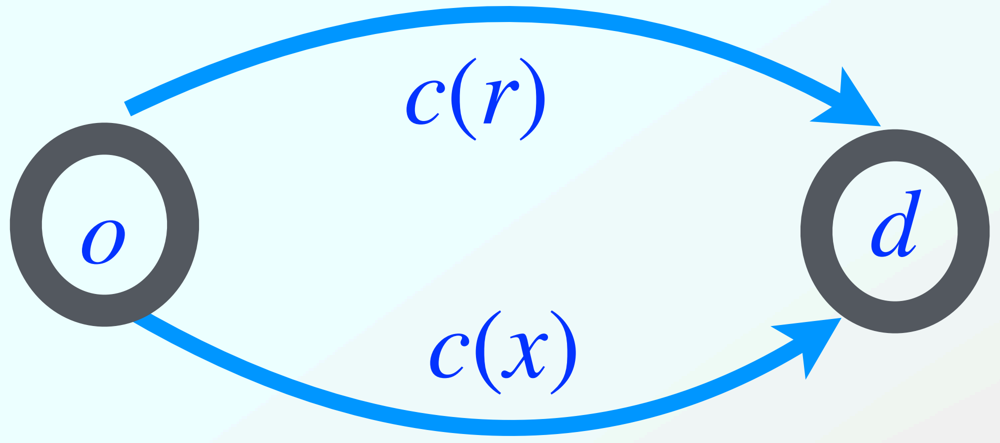


$f_{(o,v)} = f_{(w,d)} = \frac{3}{4}$ $f_{(ow)} = f_{(vd)} = \frac{1}{4}$

$f_{(vw)} = \frac{1}{2}$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r\cdot c(r)}{x\cdot c(x) + (r-x)\cdot c(r)} \right\}$

Proof sketch: Let $G=(V,E)$ be a SRN with r unit of traffic between o to d.

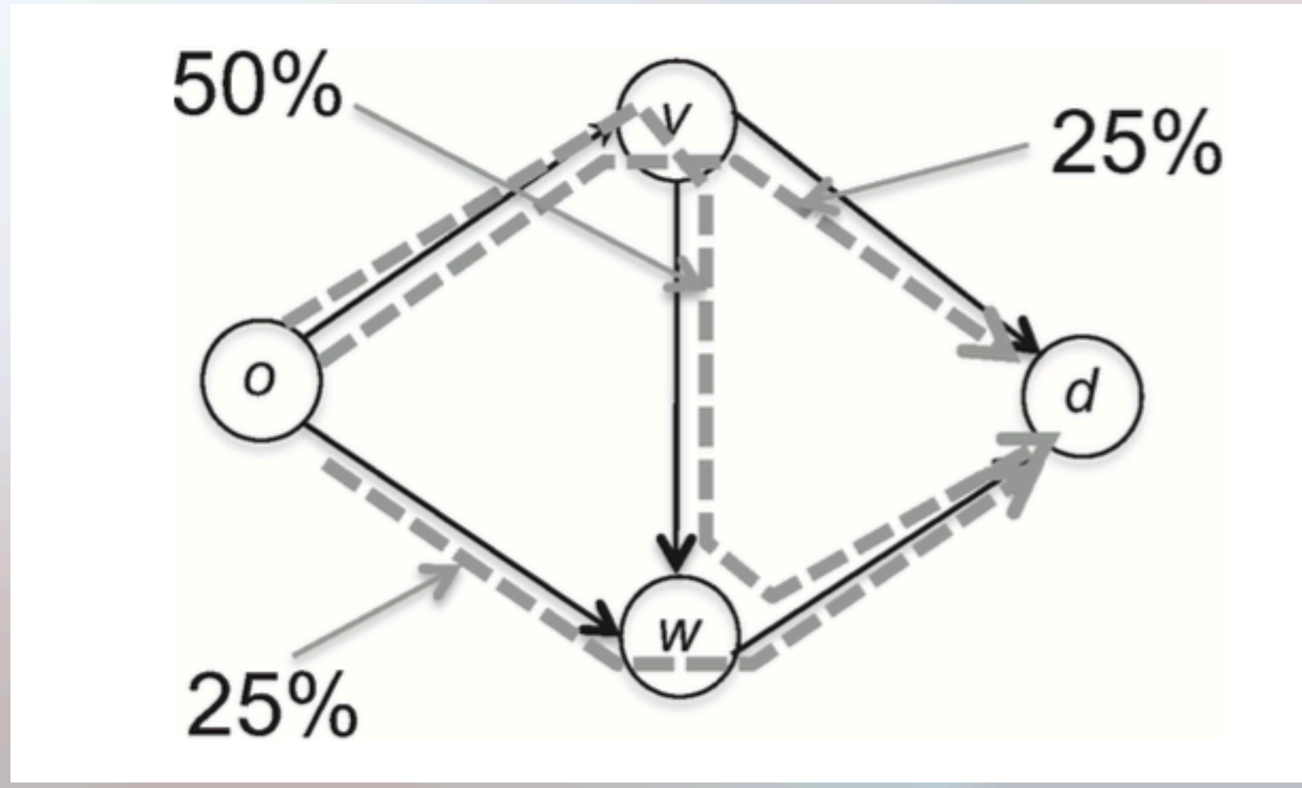


Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o-d$ paths

Non-negative vector $\{f_P\}_{P\in\mathscr{P}}$ with $\sum_{p\in\mathscr{P}} f_P = r$

For an edge $e\in E$ and flow f we write $f_e = \sum_{P\in\mathscr{P}:e\in P} f_P$



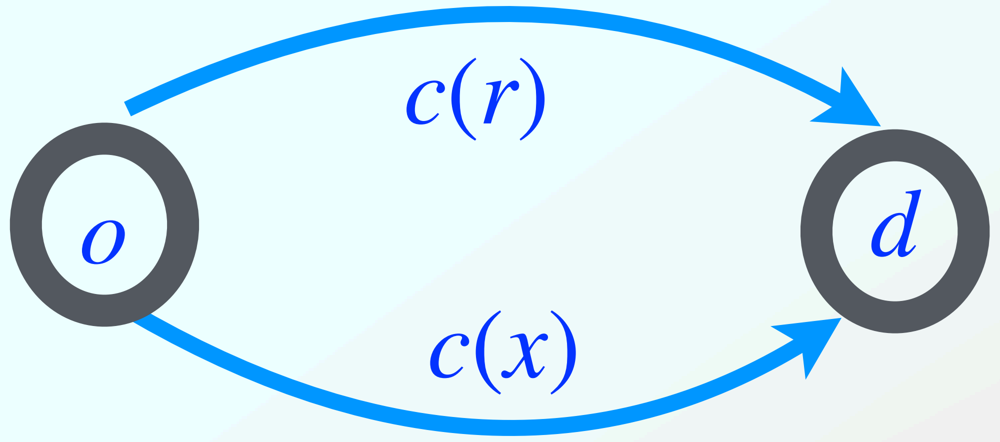
In equilibrium flow, traffic travels only along fastest $o-d$ paths induced by the flow

$$f_{(o,v)} = f_{(w,d)} = \frac{3}{4} \qquad f_{(ow)} = f_{(vd)} = \frac{1}{4}$$

$$f_{(vw)} = \frac{1}{2}$$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.

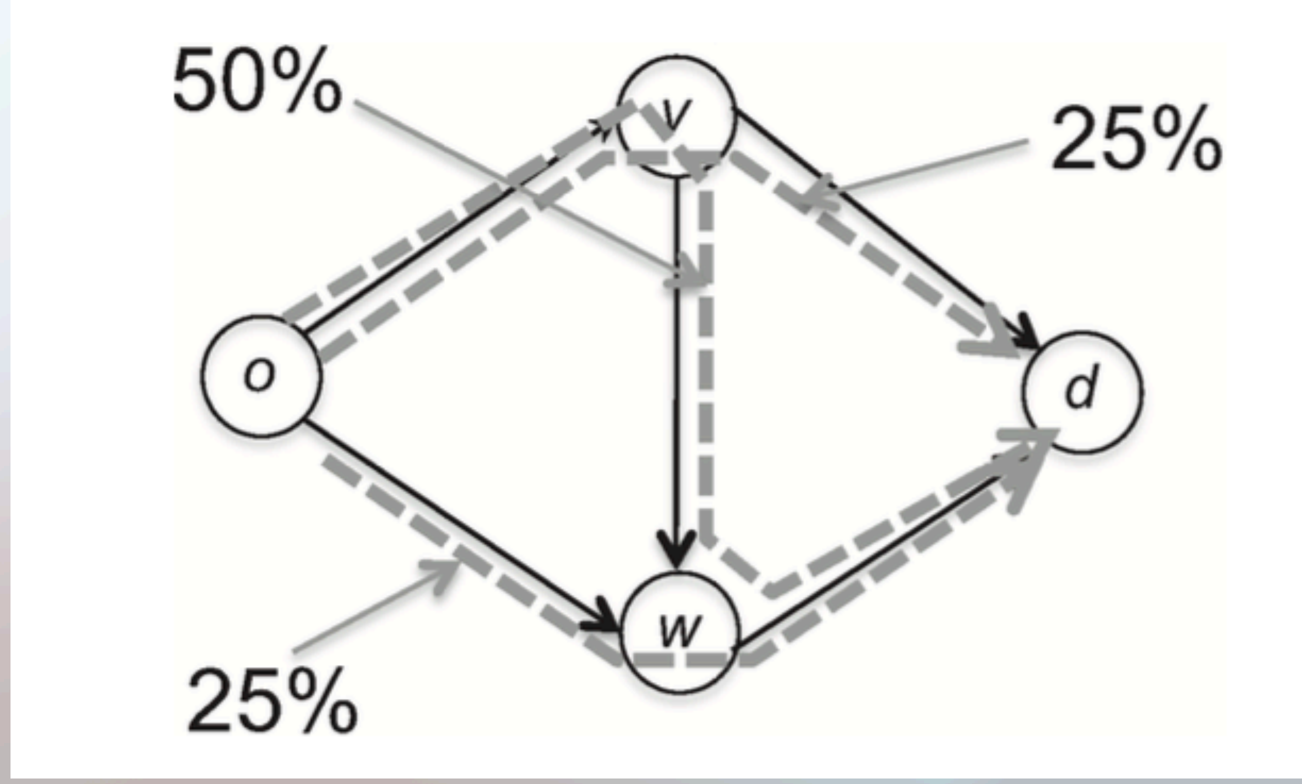


Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths

Non-negative vector $\{f_P\}_{P \in \mathscr{P}}$ with $\sum_{p \in \mathscr{P}} f_P = r$

For an edge $e \in E$ and flow f we write $f_e = \sum_{P \in \mathscr{P}: e \in P} f_P$



In equilibrium flow, traffic travels only along fastest $o - d$ paths induced by the flow

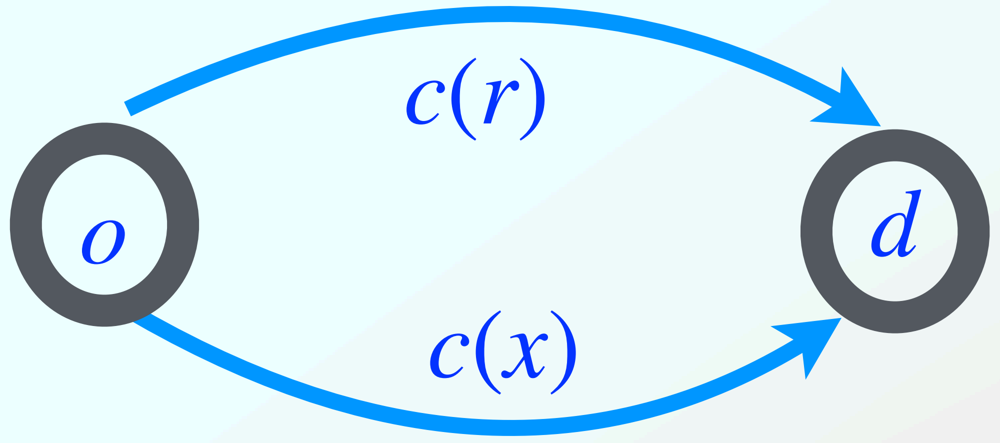
A flow f is an equilibrium if $f_{\hat{P}} > 0$ only when $\hat{P} \in \text{argmin}_{P \in \mathscr{P}} \left\{ \sum_{e \in P} c_e(f_e) \right\}$

$f_{(o,v)} = f_{(w,d)} = \frac{3}{4}$ $\quad f_{(ow)} = f_{(vd)} = \frac{1}{4}$

$f_{(vw)} = \frac{1}{2}$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

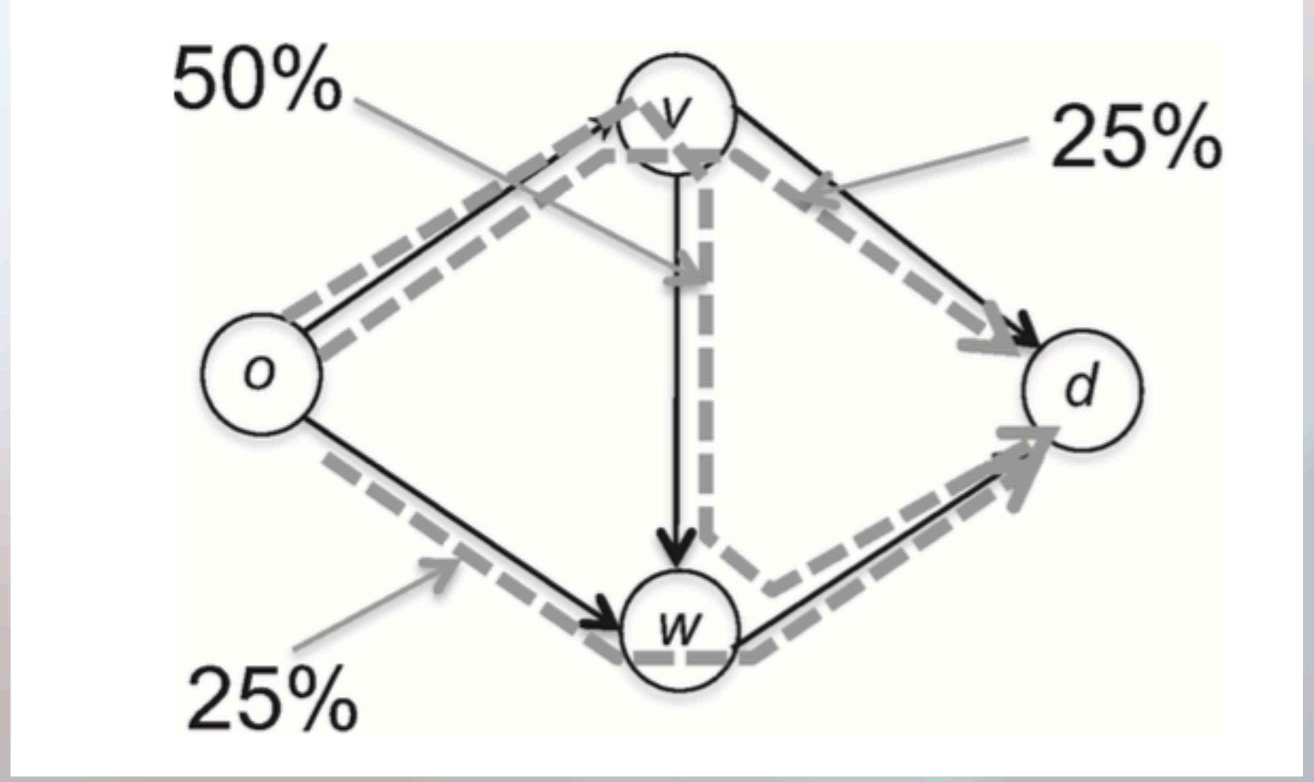
Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.



Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths Non-negative vector $\{f_P\}_{P \in \mathscr{P}}$ with $\sum_{p \in \mathscr{P}} f_P = r$

For an edge $e \in E$ and flow f we write $f_e = \sum_{P \in \mathscr{P} : e \in P} f_P$



In equilibrium flow, traffic travels only along fastest $o - d$ paths induced by the flow

A flow f is an equilibrium if $f_{\hat{P}} > 0$ only when $\hat{P} \in \text{argmin}_{P \in \mathscr{P}} \left\{ \sum_{e \in P} c_e(f_e) \right\}$

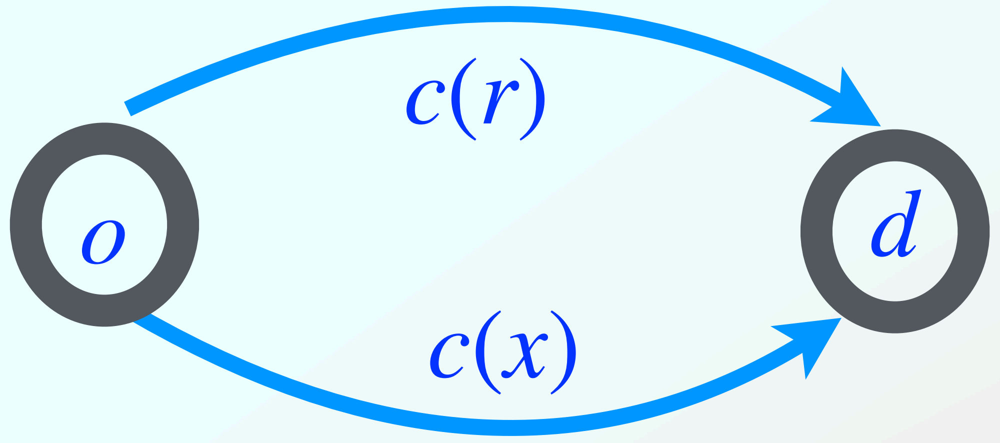
$f_{(o,v)} = f_{(w,d)} = \frac{3}{4}$ $f_{(ow)} = f_{(vd)} = \frac{1}{4}$

$f_{(vw)} = \frac{1}{2}$

NOT equilibrium flow!

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch: Let $G = (V, E)$ be a SRN with r unit of traffic between o to d.

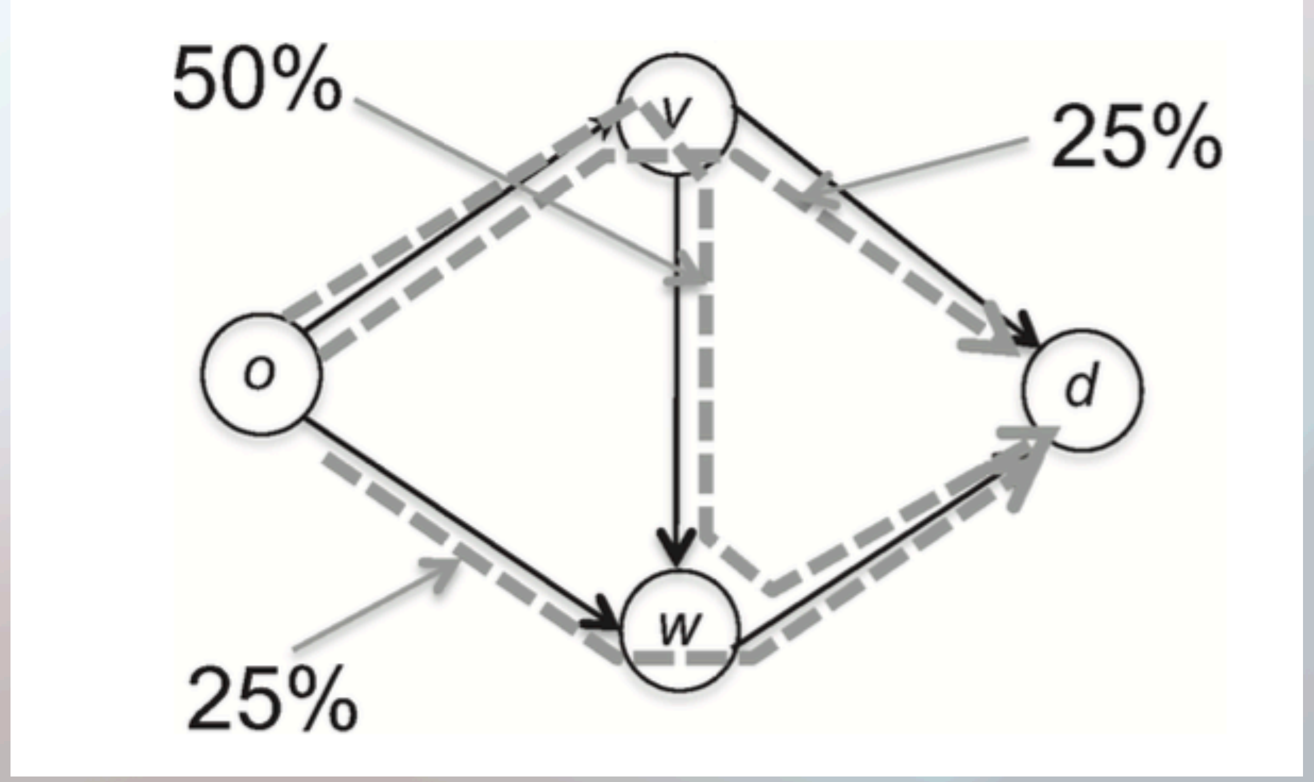


Let $\mathscr{P}$ denote the set of non-empty o-d paths of G.

A flow describes how traffic is split over the $o - d$ paths. Non-negative vector $\{f_P\}_{P \in \mathscr{P}}$ with $\sum_{p \in \mathscr{P}} f_P = r$

For an edge $e \in E$ and flow f we write $f_e = \sum_{P \in \mathscr{P} : e \in P} f_P$

In equilibrium flow, traffic travels only along fastest $o - d$ paths induced by the flow



A flow f is an equilibrium if $f_{\hat{P}} > 0$ only when $\hat{P} \in \mathrm{argmin}_{P \in \mathscr{P}} \left\{ \sum_{e \in P} c_e(f_e) \right\}$

$f_{(o,v)} = f_{(w,d)} = \frac{3}{4}$ $\quad f_{(ow)} = f_{(vd)} = \frac{1}{4}$

$f_{(vw)} = \frac{1}{2}$

NOT equilibrium flow! Shortest path is zig-zag & $f_{(o,w,d)} \neq 0$

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e \in E} c_e(f_e)$

Proof sketch (contd):

> For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e \in E} c_e(f_e)$

Thus, cost of flow f is $C(f) = \sum_{P \in \mathscr{P}} f_P \cdot c_P(f)$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e \in E} c_e(f_e)$

Thus, cost of flow f is $C(f) = \sum_{P \in \mathscr{P}} f_P \cdot c_P(f)$ $\cdots$which is same as $C(f) = \sum_{e \in E} f_e \cdot c_e(f_e)$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e \in E} c_e(f_e)$

Thus, cost of flow f is $C(f) = \sum_{P \in \mathscr{P}} f_P \cdot c_P(f)$ ⋯which is same as $C(f) = \sum_{e \in E} f_e \cdot c_e(f_e)$

Why ?

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x)\cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e\in E} c_e(f_e)$

Thus, cost of flow f is $C(f) = \sum_{P\in\mathscr{P}} f_P \cdot c_P(f)$ $\cdots$which is same as $C(f) = \sum_{e\in E} f_e \cdot c_e(f_e)$

Why ?

Let f and $f^\star$ denote equilibrium and min-cost flows in the network, respectively.

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x)\cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e\in E} c_e(f_e)$

Thus, cost of flow f is $C(f) = \sum_{P\in\mathscr{P}} f_P \cdot c_P(f)$ $\cdots$which is same as $C(f) = \sum_{e\in E} f_e \cdot c_e(f_e)$

Why ?

Let f and $f^\star$ denote equilibrium and min-cost flows in the network, respectively.

Sufficient to prove:

Proof sketch (contd):

> For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x) \cdot c(r)} \right\}$

We denote the total time taken by (i.e., cost of) flow f by $C(f)$

Define cost of f through path P is $c_P(f) = \sum_{e\in E} c_e(f_e)$

Thus, cost of flow f is $C(f) = \sum_{P\in\mathscr{P}} f_P \cdot c_P(f)$ $\cdots$which is same as $C(f) = \sum_{e\in E} f_e \cdot c_e(f_e)$

Why ?

Let f and $f^\star$ denote equilibrium and min-cost flows in the network, respectively.

Sufficient to prove: $\alpha(\mathscr{C}) \geq \frac{C(f)}{C(f^\star)} = POA$

Part 1: Freezing the cost of every edge e at equilibrium value $c_e(f_e)$ makes f optimal

Step 2: Quantify how much can $f^\star$ be better than f ?

Part 1: Freezing the cost of every edge e at equilibrium value $c_e(f_e)$ makes f optimal

Claim: $\sum_{e \in E} f_e^\star \cdot c_e(f_e) \geq \sum_{e \in E} f_e \cdot c_e(f_e)$

Step 2: Quantify how much can $f^\star$ be better than f ?

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the *POA* is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x)\cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x)\cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x)\cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging

$$f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} f_e \cdot c_e(f_e) + (f_e^\star - f_e) \cdot c_e(f_e)$$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r\cdot c(r)}{x\cdot c(x) + (r-x)\cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging

$$f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} f_e \cdot c_e(f_e) + (f_e^\star - f_e) \cdot c_e(f_e)$$

Summing over all the edges yields

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r-x) \cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging

$$f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} f_e \cdot c_e(f_e) + (f_e^\star - f_e) \cdot c_e(f_e)$$

Summing over all the edges yields

$$C(f^\star) = \sum_{e\in E} f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} \sum_{e\in E} f_e \cdot c_e(f_e) + \sum_{e\in E} (f_e^\star - f_e) \cdot c_e(f_e)$$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging
$$f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} f_e \cdot c_e(f_e) + (f_e^\star - f_e) \cdot c_e(f_e)$$

Summing over all the edges yields
$$C(f^\star) = \sum_{e \in E} f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} \sum_{e \in E} f_e \cdot c_e(f_e) + \sum_{e \in E} (f_e^\star - f_e) \cdot c_e(f_e)$$

$$C(f^\star) \geq \frac{C(f)}{\alpha(\mathscr{C})} + \sum_{e \in E} (f_e^\star - f_e) \cdot c_e(f_e) \geq \frac{C(f)}{\alpha(\mathscr{C})}$$

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c\in\mathscr{C}} \sup_{r\geq 0} \sup_{x\geq 0} \left\{ \frac{r\cdot c(r)}{x\cdot c(x) + (r-x)\cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging

$$f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} f_e \cdot c_e(f_e) + (f_e^\star - f_e)\cdot c_e(f_e)$$

Summing over all the edges yields

$$C(f^\star) = \sum_{e\in E} f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} \sum_{e\in E} f_e \cdot c_e(f_e) + \sum_{e\in E} (f_e^\star - f_e)\cdot c_e(f_e)$$

$$C(f^\star) \geq \frac{C(f)}{\alpha(\mathscr{C})} + \sum_{e\in E} (f_e^\star - f_e)\cdot c_e(f_e) \geq \frac{C(f)}{\alpha(\mathscr{C})}$$

Using claim

Proof sketch (contd):

For every set $\mathscr{C}$ of cost function and every selfish routing network with cost function in $\mathscr{C}$, the POA is at most $\alpha(\mathscr{C})$, where $\alpha(\mathscr{C}) = \sup_{c \in \mathscr{C}} \sup_{r \geq 0} \sup_{x \geq 0} \left\{ \frac{r \cdot c(r)}{x \cdot c(x) + (r - x) \cdot c(r)} \right\}$

Think about : POA across an edge $e \in E$

Replace c with c_e, flow f_e for r, and $f_e^\star$ for x

Since $\alpha(\mathscr{C})$ is supremum over all choices of c, r, and x, so over $(c_e, f_e, f_e^\star)$ as well

By rearranging

$$f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} f_e \cdot c_e(f_e) + (f_e^\star - f_e) \cdot c_e(f_e)$$

Summing over all the edges yields

$$C(f^\star) = \sum_{e \in E} f_e^\star \cdot c_e(f_e^\star) \geq \frac{1}{\alpha(\mathscr{C})} \sum_{e \in E} f_e \cdot c_e(f_e) + \sum_{e \in E} (f_e^\star - f_e) \cdot c_e(f_e)$$

$$C(f^\star) \geq \frac{C(f)}{\alpha(\mathscr{C})} + \sum_{e \in E} (f_e^\star - f_e) \cdot c_e(f_e) \geq \frac{C(f)}{\alpha(\mathscr{C})}$$

Using claim QED