Structure and stability of equilibria in a queue-based traffic model

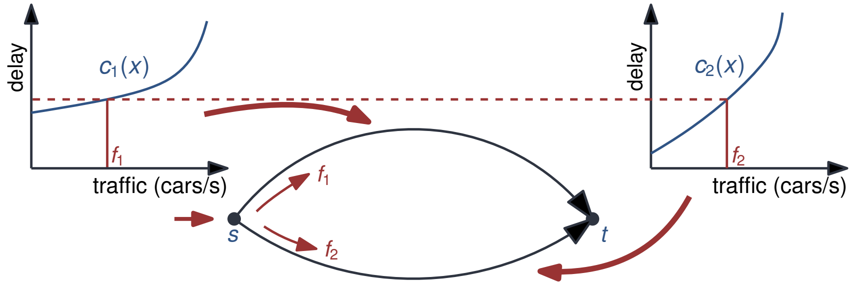
Neil Olver

LSE

Dutch Seminar on Optimization, September 2024

Static models of traffic

Static models well-studied from the algorithmic game theory perspective



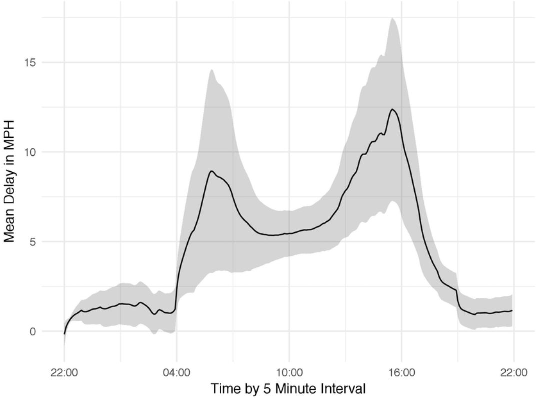
- Equilibria computable via a convex program *Beckmann et al. '56*
- Price of anarchy bounds *Roughgarden & Tardos, ...*
- Braess's paradox
- ...

Dynamic aspects can be very important

Static models can be useful, but they do miss something...

Dynamic aspects can be very important

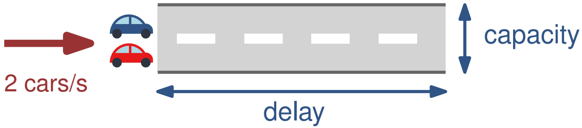
Static models can be useful, but they do miss something...



Credit: Brent, Beland (2020). Traffic congestion, transportation policies, and the performance of first responders, J. Environmental Economics and Management.

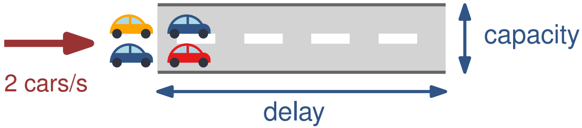
The Vickrey bottleneck model

Vickrey '69



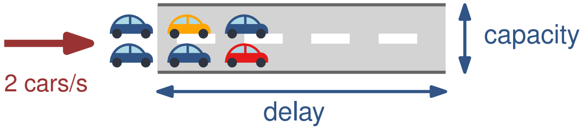
The Vickrey bottleneck model

Vickrey '69



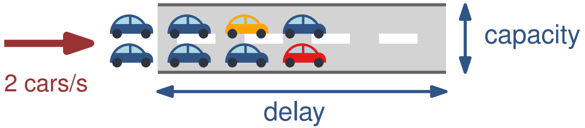
The Vickrey bottleneck model

Vickrey '69



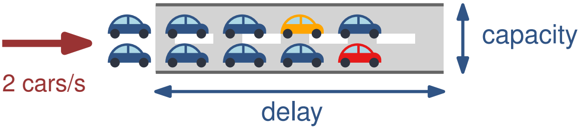
The Vickrey bottleneck model

Vickrey '69



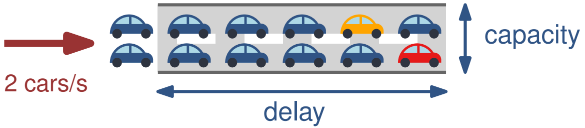
The Vickrey bottleneck model

Vickrey '69



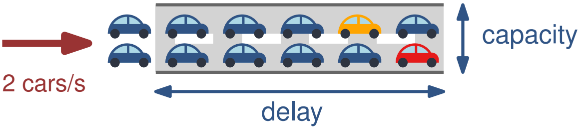
The Vickrey bottleneck model

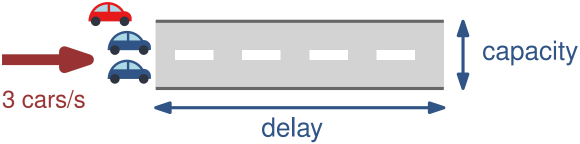
Vickrey '69



The Vickrey bottleneck model

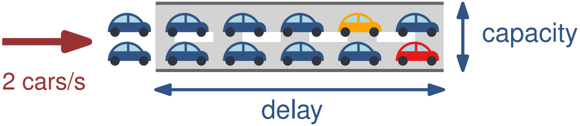
Vickrey '69

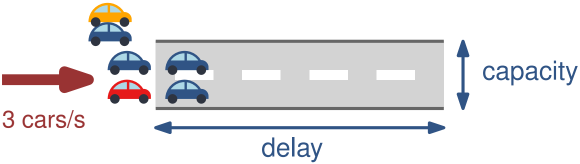




The Vickrey bottleneck model

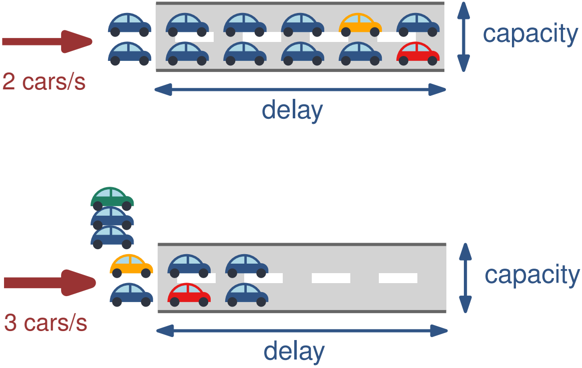
Vickrey '69





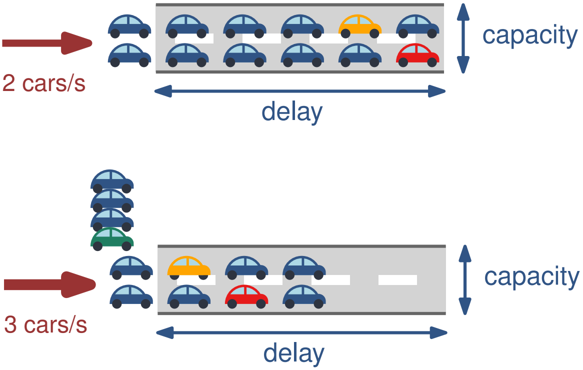
The Vickrey bottleneck model

Vickrey '69



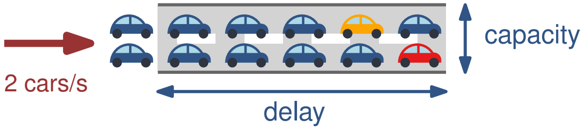
The Vickrey bottleneck model

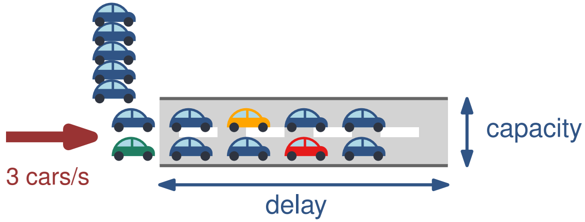
Vickrey '69



The Vickrey bottleneck model

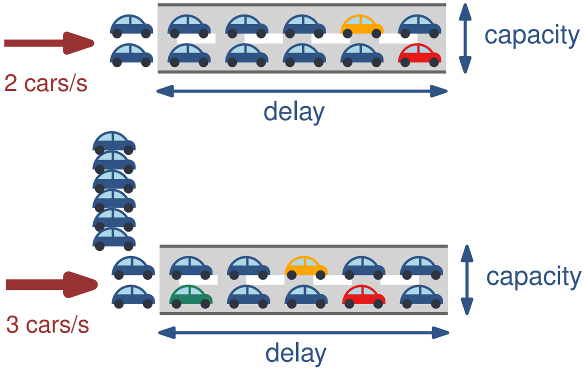
Vickrey '69





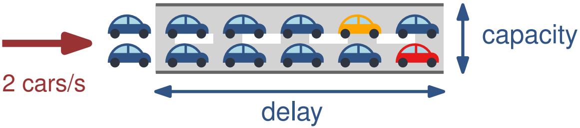
The Vickrey bottleneck model

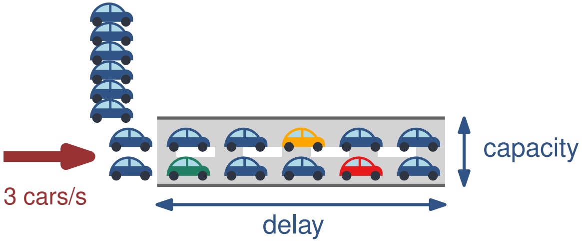
Vickrey '69



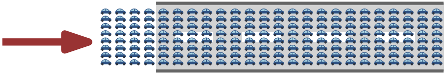
The Vickrey bottleneck model

Vickrey '69

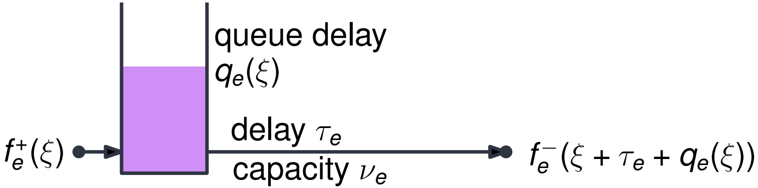




- Continuous, nonatomic limit



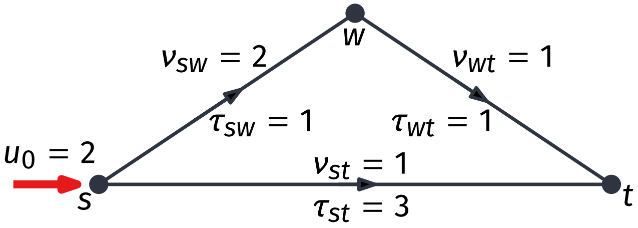
The Vickrey bottleneck model



$$\frac{dq_e(\xi)}{d\xi} = \begin{cases} \frac{1}{\nu_e}[f_e^+(\xi) - \nu_e] & \text{if } q_e(\xi) > 0 \\ \frac{1}{\nu_e}[f_e^+(\xi) - \nu_e]^+ & \text{if } q_e(\xi) = 0 \end{cases}$$

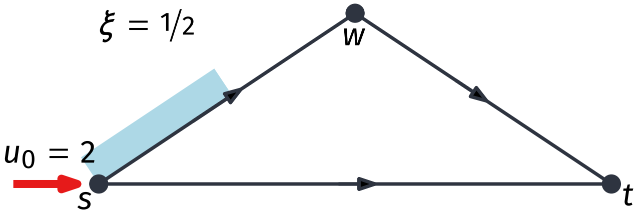
Networks of Vickrey bottlenecks

- Each link behaves as per the Vickrey bottleneck model: (f_e^+, f_e^-, z_e).
- Flow conservation: except for s, t, flow in = flow out at all times.
- All traffic from s to t; constant inflow rate u_0 from time 0.



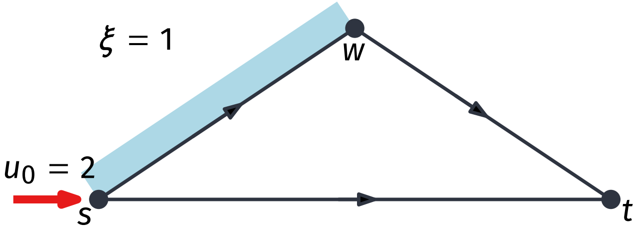
Networks of Vickrey bottlenecks

- Each link behaves as per the Vickrey bottleneck model: (f_e^+, f_e^-, z_e).
- Flow conservation: except for s, t, flow in = flow out at all times.
- All traffic from s to t; constant inflow rate u_0 from time 0.



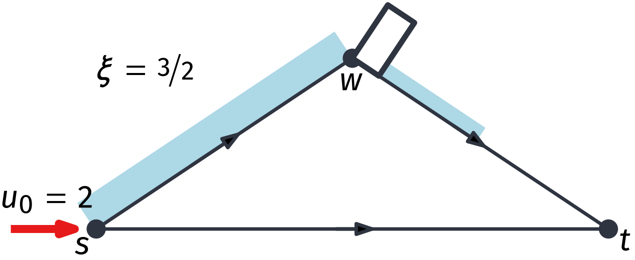
Networks of Vickrey bottlenecks

- Each link behaves as per the Vickrey bottleneck model: (f_e^+, f_e^-, z_e).
- Flow conservation: except for s, t, flow in = flow out at all times.
- All traffic from s to t; constant inflow rate u_0 from time 0.



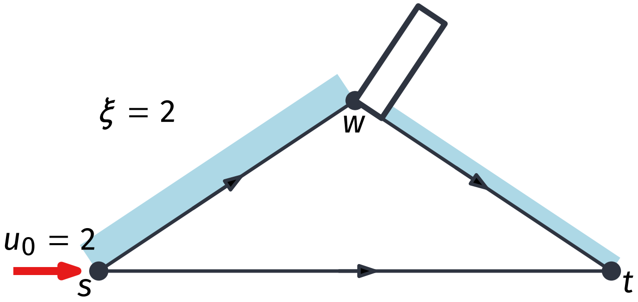
Networks of Vickrey bottlenecks

- Each link behaves as per the Vickrey bottleneck model: (f_e^+, f_e^-, z_e).
- Flow conservation: except for s, t, flow in $=$ flow out at all times.
- All traffic from s to t; constant inflow rate u_0 from time 0.

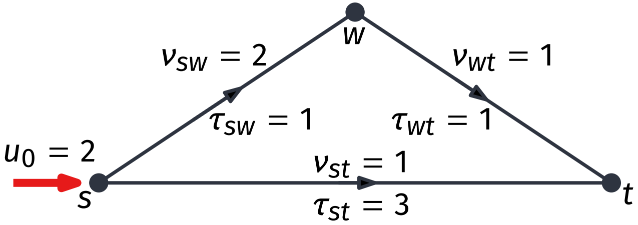


Networks of Vickrey bottlenecks

- Each link behaves as per the Vickrey bottleneck model: (f_e^+, f_e^-, z_e).
- Flow conservation: except for s, t, flow in = flow out at all times.
- All traffic from s to t; constant inflow rate u_0 from time 0.

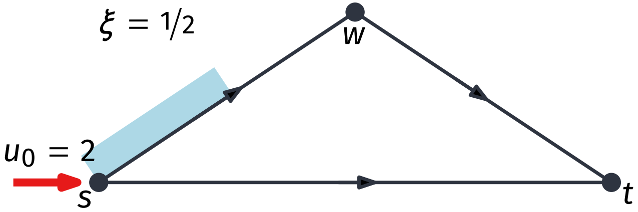


Equilibria in networks of Vickrey bottlenecks



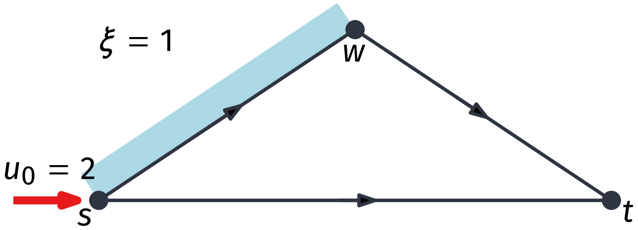
- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Equilibria in networks of Vickrey bottlenecks



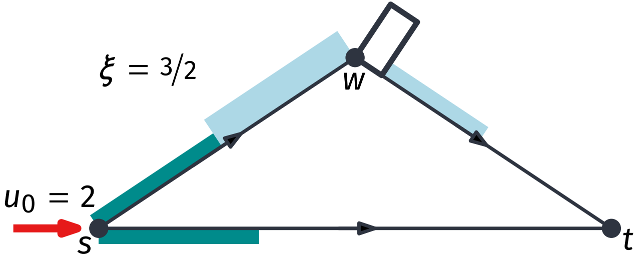
- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Equilibria in networks of Vickrey bottlenecks



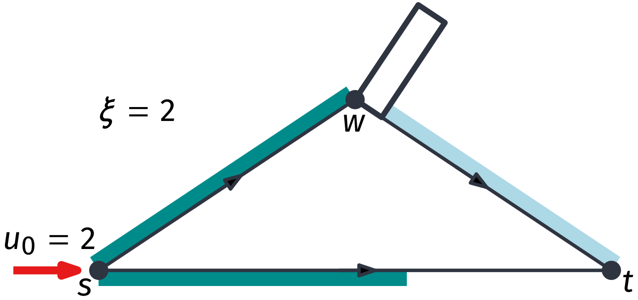
- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Equilibria in networks of Vickrey bottlenecks



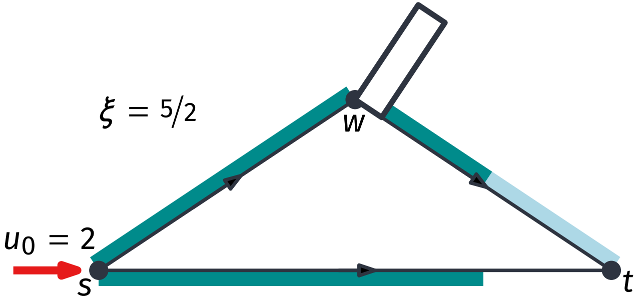
- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Equilibria in networks of Vickrey bottlenecks



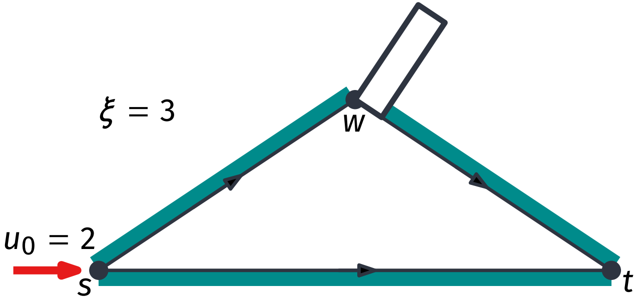
- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Equilibria in networks of Vickrey bottlenecks



- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Equilibria in networks of Vickrey bottlenecks



- Dynamic equilibria: users choose routes to arrive as early as possible, given congestion (queues) induced by other users.
- In the sense of Nash: no deviating improvement possible in hindsight.

Relevance

- Model is a continuous approximation of a discrete reality
- Travel times on roads are likely to be noisy
- Users may take only approximately shortest routes, not precisely shortest routes
- ...

Are dynamic equilibria “stable” under perturbations?

Do they have anything to do with equilibria in models that are “almost” the same?

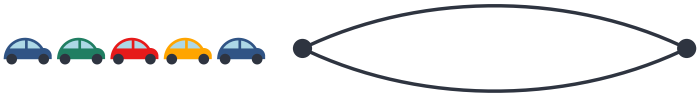
A more precise question (1)

One can define discrete (packet) versions of the deterministic queueing model. *Hoefer et al. '11, Werth et al. '14, ...*

Equilibrium: no packet can arrive to the sink earlier using a different route.

Suppose we fix an instance, but divide up the flow into packets of smaller and smaller size.

The hope: as packet size goes to 0, equilibrium converges to the dynamic equilibrium of the deterministic queueing model.



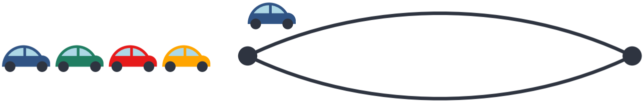
A more precise question (1)

One can define discrete (packet) versions of the deterministic queueing model. *Hoefer et al. '11, Werth et al. '14, …*

Equilibrium: no packet can arrive to the sink earlier using a different route.

Suppose we fix an instance, but divide up the flow into packets of smaller and smaller size.

The hope: as packet size goes to 0, equilibrium converges to the dynamic equilibrium of the deterministic queueing model.



A more precise question (1)

One can define discrete (packet) versions of the deterministic queueing model. *Hoefer et al. '11, Werth et al. '14, ...*

Equilibrium: no packet can arrive to the sink earlier using a different route.

Suppose we fix an instance, but divide up the flow into packets of smaller and smaller size.

The hope: as packet size goes to 0, equilibrium converges to the dynamic equilibrium of the deterministic queueing model.

A more precise question (1)

One can define discrete (packet) versions of the deterministic queueing model. *Hoefer et al. '11, Werth et al. '14, . . .*

Equilibrium: no packet can arrive to the sink earlier using a different route.

Suppose we fix an instance, but divide up the flow into packets of smaller and smaller size.

The hope: as packet size goes to 0, equilibrium converges to the dynamic equilibrium of the deterministic queueing model.

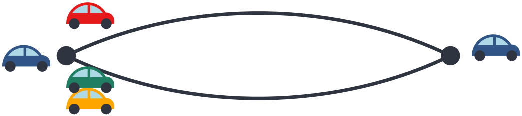
A more precise question (1)

One can define discrete (packet) versions of the deterministic queueing model. *Hoefer et al. ’11, Werth et al. ’14, …*

Equilibrium: no packet can arrive to the sink earlier using a different route.

Suppose we fix an instance, but divide up the flow into packets of smaller and smaller size.

The hope: as packet size goes to 0, equilibrium converges to the dynamic equilibrium of the deterministic queueing model.



A more precise question (1)

One can define discrete (packet) versions of the deterministic queueing model. *Hoefer et al. '11, Werth et al. '14, ...*

Equilibrium: no packet can arrive to the sink earlier using a different route.

Suppose we fix an instance, but divide up the flow into packets of smaller and smaller size.

The hope: as packet size goes to 0, equilibrium converges to the dynamic equilibrium of the deterministic queueing model.

A more precise question (2)

An ϵ-equilibrium is a joint strategy choice of all users in which each user arrives at the sink at most ϵ later than the earliest possible, given the delays caused by other users.

Fix an instance. Let $\epsilon_1, \epsilon_2, \ldots$ be a sequence converging to 0. Let ϕ_i be an ϵ_i-equilibrium for each i.

The hope: ϕ_i converges to the exact dynamic equilibrium as $i \longrightarrow \infty$.

Main theorem *O.-Sering-Vargas Koch FOCS '23*

(Informal.) In both of these situations (among others), convergence to the dynamic equilibrium is guaranteed.

- Point in favour of "meaningfulness" of the equilibrium concept
- Allows for results in the deterministic queueing model to be ported to other models
  - If the network capacity is at least as large as the inflow, queues stay bounded in dynamic equilibria

    Correa-Cominetti-O. '17, '22
  - So the same holds for packet models, for sufficiently small packet sizes
- Shows that discretization can be used to compute approximate equilibria in the nonatomic model

Labels and equilibrium conditions

- Agent set $\mathcal{A}$. Agent $a \in \mathcal{A}$ departs source at time ϑ_a.
- Strategy profile $\varphi : \mathcal{A} \to \mathcal{P} = \{s\text{-}t \text{ paths}\}$.

  $\Longrightarrow$ a flow $x'(\theta)$ of value u_0 describing what particles departing at time θ do.

Labels and equilibrium conditions

- Agent set $\mathcal{A}$. Agent $a \in \mathcal{A}$ departs source at time ϑ_a.
- Strategy profile $\varphi : \mathcal{A} \to \mathcal{P} = \{s\text{-}t \text{ paths}\}$.

  $\Longrightarrow$ a flow $x'(\theta)$ of value u_0 describing what particles departing at time θ do.

Network loading: Maps φ to the resulting flow-over-time $\big(f_e^+(\cdot), f_e^-(\cdot), z_e(\cdot)\big)_{e \in E}$.

Labels and equilibrium conditions

- Agent set $\mathcal{A}$. Agent $a \in \mathcal{A}$ departs source at time ϑ_a.
- Strategy profile $\varphi : \mathcal{A} \to \mathcal{P} = \{s\text{-}t \text{ paths}\}$.

  $\Longrightarrow$ a flow $x'(\theta)$ of value u_0 describing what particles departing at time θ do.

Network loading: Maps φ to the resulting flow-over-time $\left(f_e^+(\cdot), f_e^-(\cdot), z_e(\cdot)\right)_{e \in E}$.

From this, we can determine:

- Departure times: for $v \in \varphi(a)$, $d_v(a)$ is the time that agent a departs node v.
- Earliest arrival labels: $\ell_v(\theta)$ is the earliest time a particle leaving s at time θ can arrive at v, taking into account queues caused by earlier particles.

Dynamic equilibrium conditions

A strategy profile φ is an equilibrium if

$$d_v(a) = \ell_v(\vartheta_a) \qquad \text{for all } a \in \mathcal{A}, v \in \varphi(a).$$

the labels $\ell_v(\theta)$ then define an equilibrium trajectory.

Dynamic equilibrium conditions

A strategy profile φ is an equilibrium if

$$d_v(a) = \ell_v(\vartheta_a) \qquad \text{for all } a \in \mathcal{A}, v \in \varphi(a).$$

the labels $\ell_v(\theta)$ then define an equilibrium trajectory.

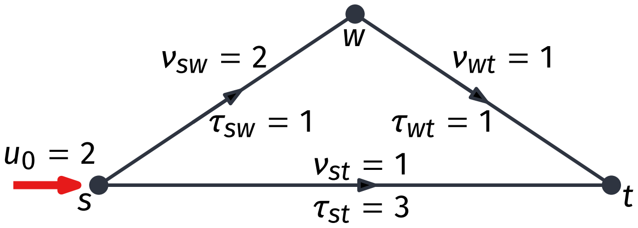
Active arcs:

$$E'_\theta := \{e = vw \in E : \ell_w(\theta) = \ell_v(\theta) + \tau_e + \underbrace{z_e(\ell_v(\theta))/\nu_e}_{q_e(\theta)}\}$$

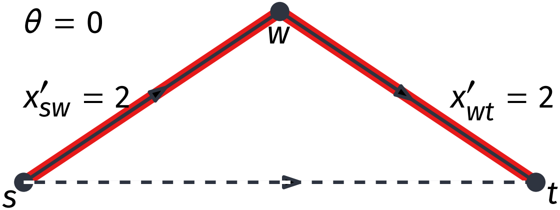
Dynamic equilibrium conditions (alternative)

$$x'_e(\theta) > 0 \quad \Rightarrow e \in E'_\theta \qquad \text{for all } \theta, e.$$

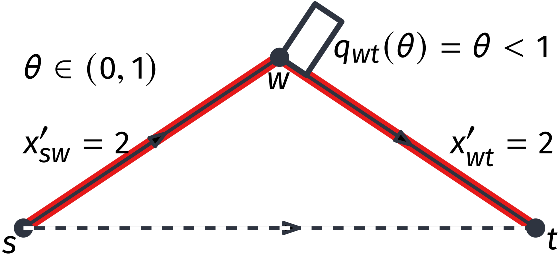
Example redux



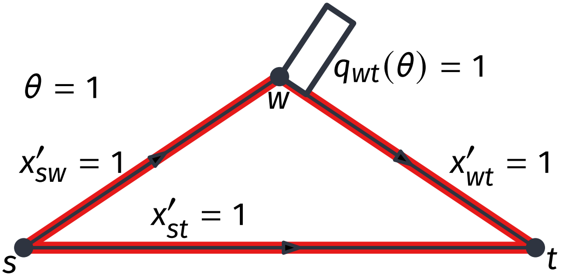
Example redux



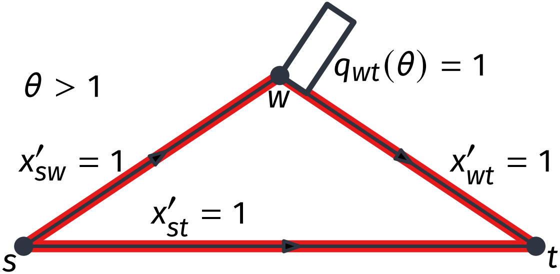
Example redux



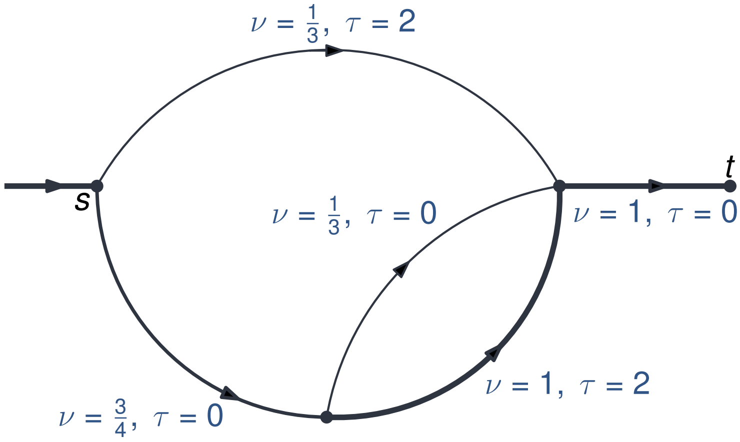
Example redux



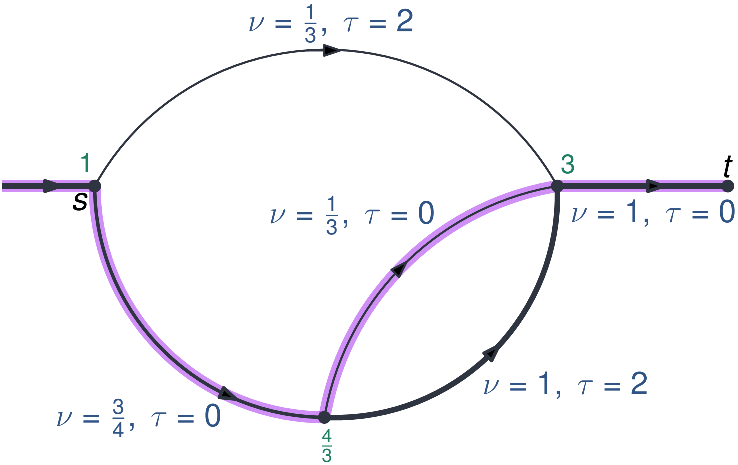
Example redux



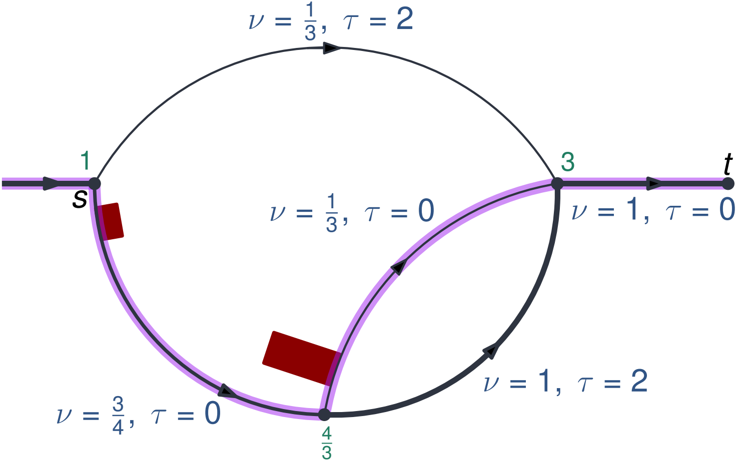
A more surprising example



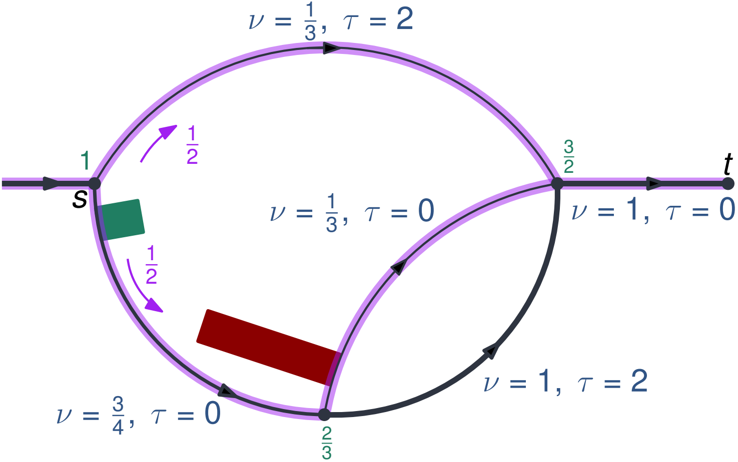
A more surprising example



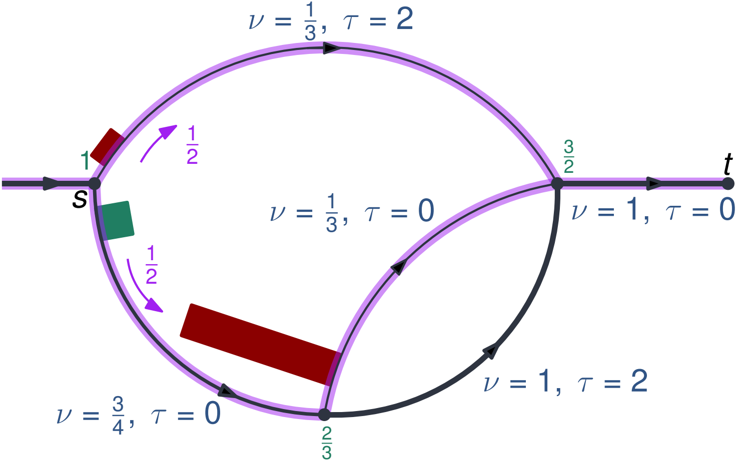
A more surprising example



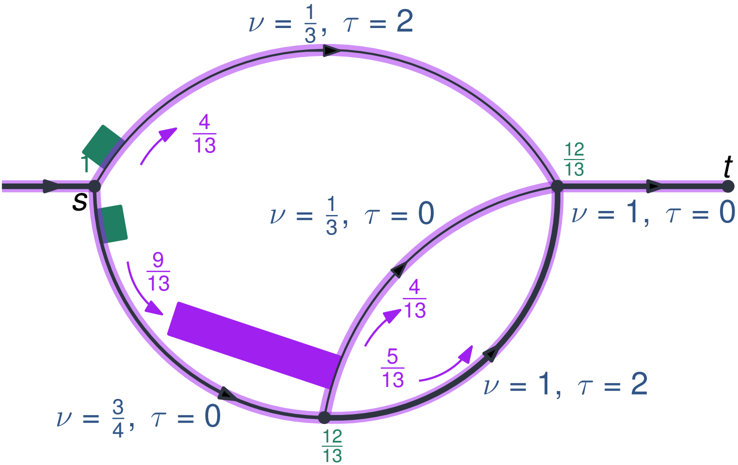
A more surprising example



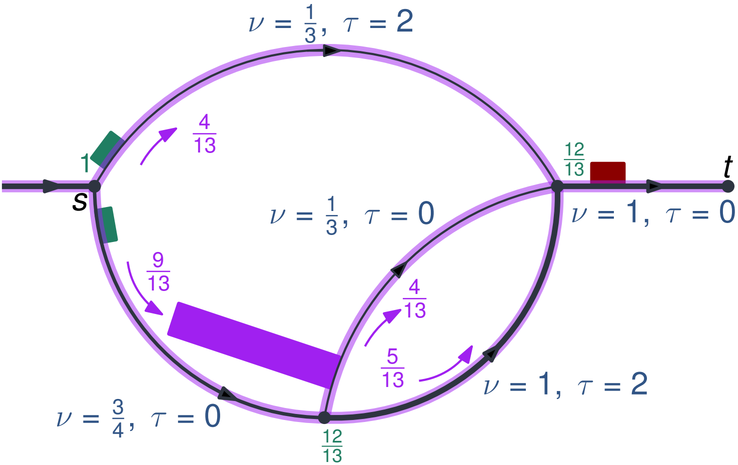
A more surprising example



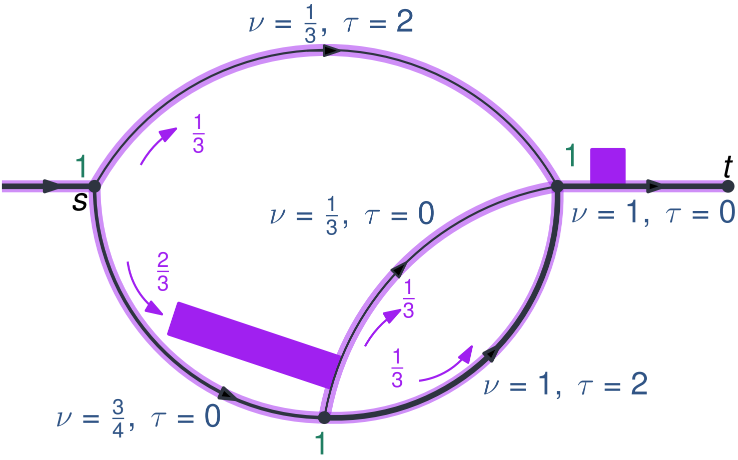
A more surprising example



A more surprising example



A more surprising example



Equilibrium structure

Koch-Skutella '11

Labels are primary.
To understand how the equilibrium develops, it suffices to keep track of the labels $\ell(\theta)$.

Equilibrium structure

Koch-Skutella '11

Labels are primary.
To understand how the equilibrium develops, it suffices to keep track of the labels $\ell(\theta)$.

- Labels suffice to determine the set of active arcs and the (relevant) queue lengths: for $e = vw$,

$$e \in E'_\theta \quad \text{if and only if} \quad \ell_w(\theta) \geq \ell_v(\theta) + \tau_e$$

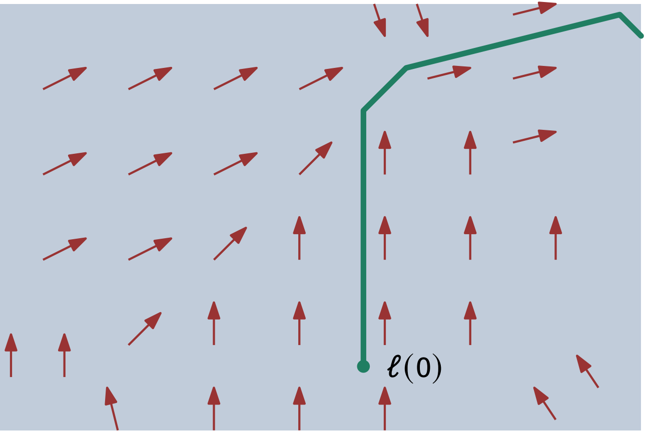
$$q_e(\theta) = [\ell_w(\theta) - \ell_v(\theta) - \tau_e]^+$$

Equilibrium structure

Koch-Skutella '11

There is a vector field $Z : \mathbb{R}^V \longrightarrow \mathbb{R}^V$ s.t. for any equilibrium,

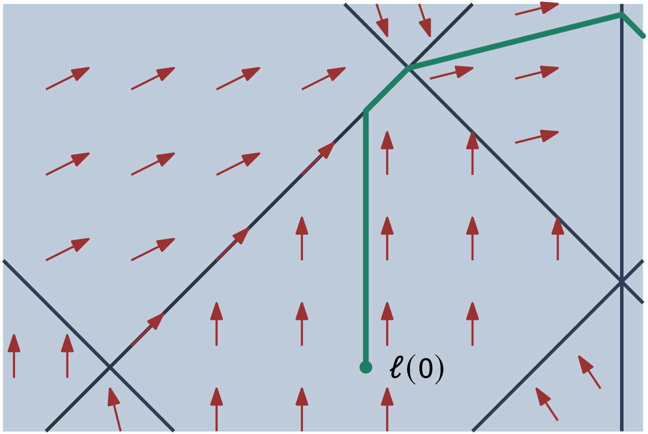
$$\ell'(\theta) = Z(\ell(\theta)) \quad \text{for almost every } \theta.$$



Equilibrium structure

Further, we can write $Z(\ell(\theta)) := Z(E'_{\ell(\theta)}, E^*_{\ell(\theta)})$, where

$$E'_l := \{e = vw \in E : l_w - l_v \geq \tau_e\}$$
$$E^*_l := \{e = vw \in E : l_w - l_v > \tau_e\}$$



Equilibrium structure

- $Z(l)$ is defined as a solution to a certain nonlinear system of equations (the “thin flow equations”) *Koch-Skutella ’11*
- This system always has a unique solution (so Z is well-defined) *Cominetti-Correa-Larré ’16*

Equilibrium structure

- $Z(l)$ is defined as a solution to a certain nonlinear system of equations (the “thin flow equations”) *Koch-Skutella ’11*
- This system always has a unique solution (so Z is well-defined) *Cominetti-Correa-Larré ’16*

- Implies the existence of an equilibrium. *KS ’11 + CCL ’16*
- $\ell(\theta)$ is piecewise-linear; we call each linear segment a phase.

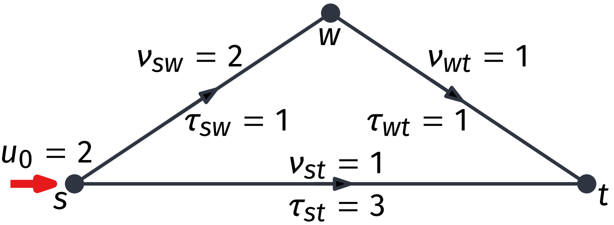
Equilibrium structure

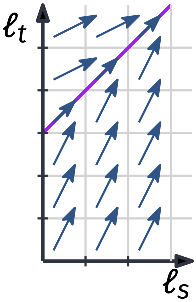
- $Z(l)$ is defined as a solution to a certain nonlinear system of equations (the "thin flow equations") *Koch-Skutella '11*
- This system always has a unique solution (so Z is well-defined) *Cominetti-Correa-Larré '16*

- Implies the existence of an equilibrium. *KS '11 + CCL '16*
- $\ell(\theta)$ is piecewise-linear; we call each linear segment a phase.

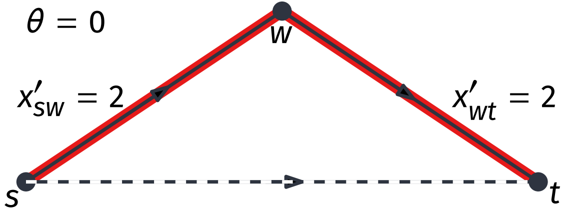
We don't know if $Z(\cdot)$ can be efficiently computed.

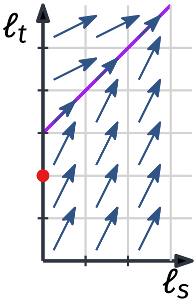
Example again



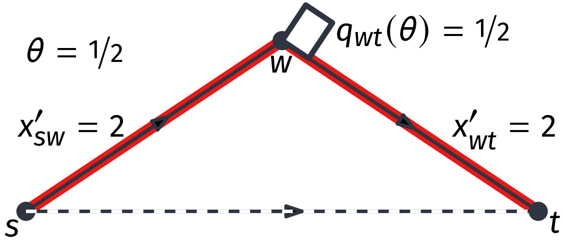


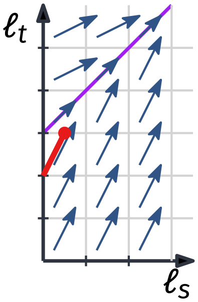
Example again



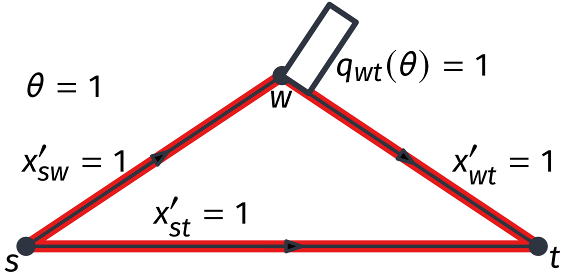


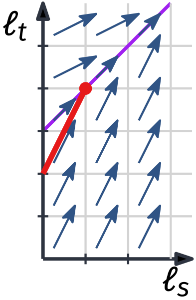
Example again



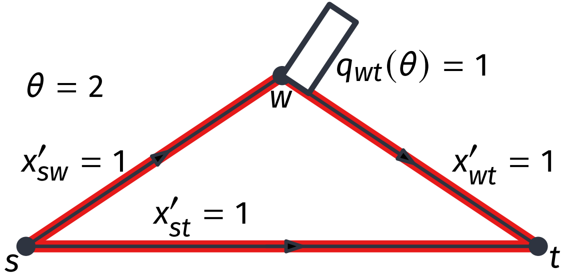


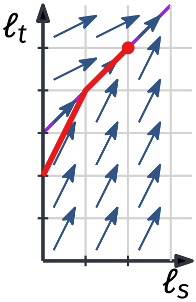
Example again



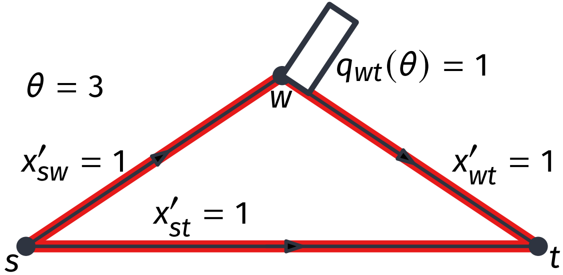


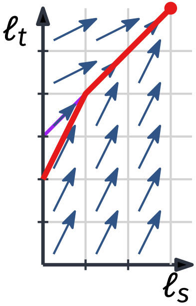
Example again





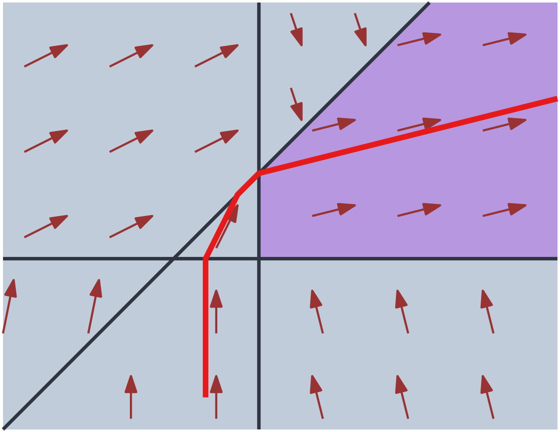
Example again





Long-term behaviour

Q: Does an equilibrium always reach a steady state, after which ℓ is linear?



Long-term behaviour

Q: Does an equilibrium always reach a steady state, after which ℓ is linear?

Theorem *O.-Sering-Vargas Koch '21*

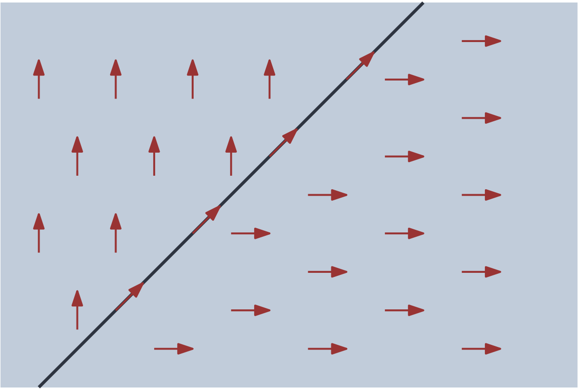
A steady state is always reached in finite time.

- Builds on Cominetti-Correa-O. (2017, 2021), which shows this under the condition that the capacity of the network is at least u_0.

  Implies bounded queues in this case.
- Key is the construction of a (rather non-obvious) non-decreasing potential.

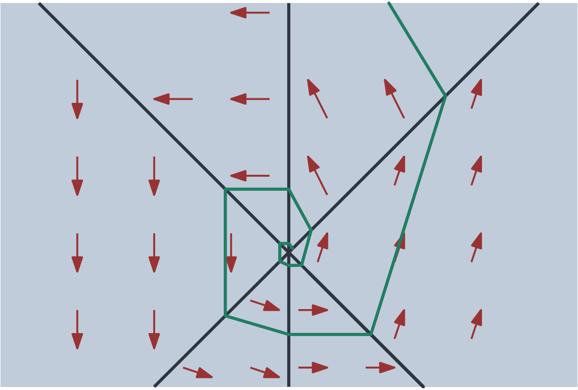
Uniqueness and continuity

Uniqueness of $Z(\cdot)$ does not imply uniqueness of ℓ.



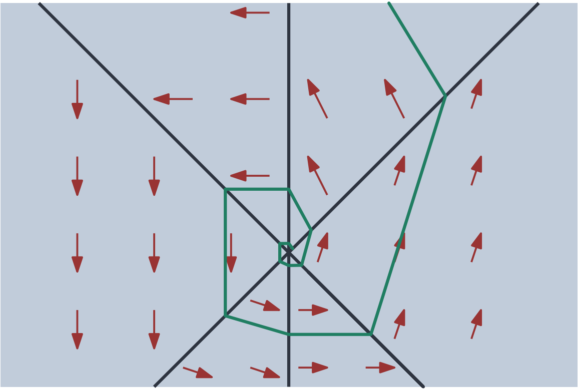
Uniqueness and continuity

Uniqueness of $Z(\cdot)$ does not imply uniqueness of ℓ.



Uniqueness and continuity

Uniqueness of $Z(\cdot)$ does not imply uniqueness of ℓ.



Theorem *O.-Sering-Vargas Koch '21*

Equilibrium trajectories are unique and depend continuously on initial conditions $\ell(0)$.

The vector fields that describe equilibria dynamics are very special!

Back to stability

Main theorem *O.-Sering-Vargas Koch FOCS '23*

(Informal.) For packet-based models (as packet size goes to zero) and for ϵ-equilibria (as $\epsilon \longrightarrow 0$), convergence to the dynamic equilibrium is guaranteed.

Strict δ-equilibria

Exact equilibria

A strategy profile φ is an exact equilibrium if

$$d_v(a) = \ell_v(\vartheta_a)) \qquad \text{for all } a \in \mathcal{A}, v \in \varphi(a);$$

the labels $\ell_v(\theta)$ then define an equilibrium trajectory.

Strict δ-equilibria

A strategy profile $\tilde{\varphi}$ is a strict δ-equilibrium if

$$\tilde{d}_v(a) \leq \tilde{\ell}_v(\vartheta_a) + \delta \qquad \text{for all } a \in \mathcal{A}, v \in \tilde{\varphi}(a);$$

the labels $\tilde{\ell}_v(\theta)$ then define a δ-trajectory.

- An ϵ-approximate equilibrium is a strict ϵ-equilibrium (but not conversely).

Formal theorem statement

Theorem

- ϵ-equilibria are strict $O(\epsilon)$-equilibria.
- Packet equilibria with packets of size ϵ are strict $O(\epsilon)$-equilibria

Main theorem *O.-Sering-Vargas Koch FOCS '23*

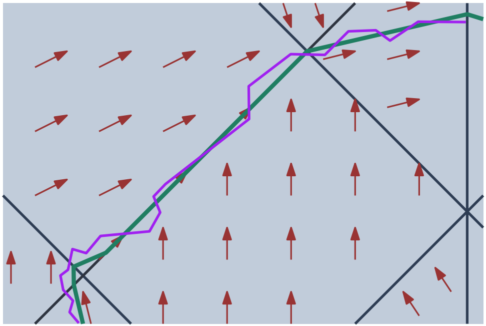
Strict δ-equilibria converge to exact dynamic equilibria as $\delta \longrightarrow 0$.

More precisely: if $\ell(\theta)$ is an equilibrium trajectory, and $\tilde{\ell}^{(i)}(\theta)$ a $\delta^{(i)}$-trajectory for each i, with $\delta^{(i)} \longrightarrow 0$, then

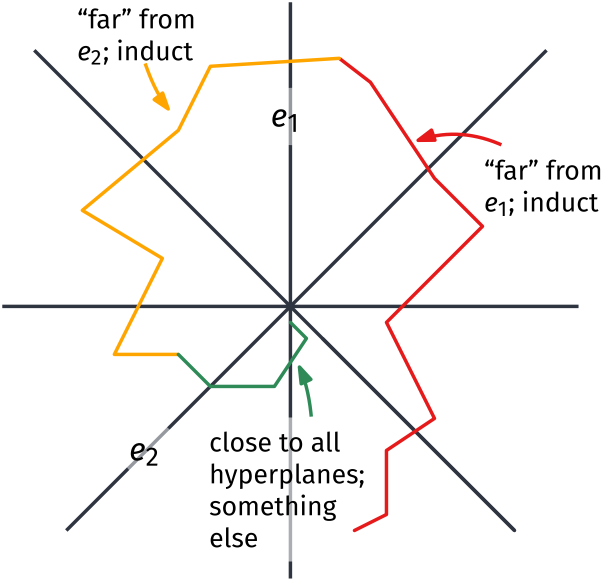
$$\sup_{\theta \geq 0} \|\ell(\theta) - \tilde{\ell}^{(i)}(\theta)\| \longrightarrow 0.$$

Continuity vs stability

- Our continuity result shows that equilibria are stable under a single perturbation (or finite number of perturbations).
- Clearly necessary, but not nearly enough:
  - $\tilde{\ell}$ need not follow the vector field anywhere.
  - A slow drift away is not acceptable; something must "pull $\tilde{\ell}$ back".

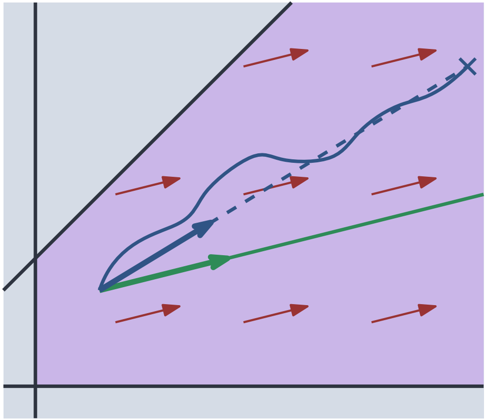


Proof heavily exploits induction on the number of hyperplanes.



Base case: equilibrium trajectory is in "steady state": all labels and queues change linearly forever.

We give a "robust" version of proof by Cominetti, Correa and Larré that $Z(\cdot)$ is unique.



Conclusion

Theorem *O.-Sering-Vargas Koch '23*

Strict δ-equilibria converge to exact dynamic equilibria as $\delta \longrightarrow 0$.

- Dependence on δ is horrible...
  Can it be shown that $\sup_\theta \|\ell(\theta) - \tilde{\ell}(\theta)\| = O(\delta)$?

Many basic open questions about equilibria remain:

- Computational complexity of computing $X(\cdot)$
- Price of anarchy
- Structure of equilibria with multiple origin-destination pairs
- ...

Conclusion

Theorem *O.-Sering-Vargas Koch '23*

Strict δ-equilibria converge to exact dynamic equilibria as $\delta \longrightarrow 0$.

- Dependence on δ is horrible...
  Can it be shown that $\sup_\theta \|\ell(\theta) - \tilde{\ell}(\theta)\| = O(\delta)$?

Many basic open questions about equilibria remain:

- Computational complexity of computing $X(\cdot)$
- Price of anarchy
- Structure of equilibria with multiple origin-destination pairs
- ...

Thank you!