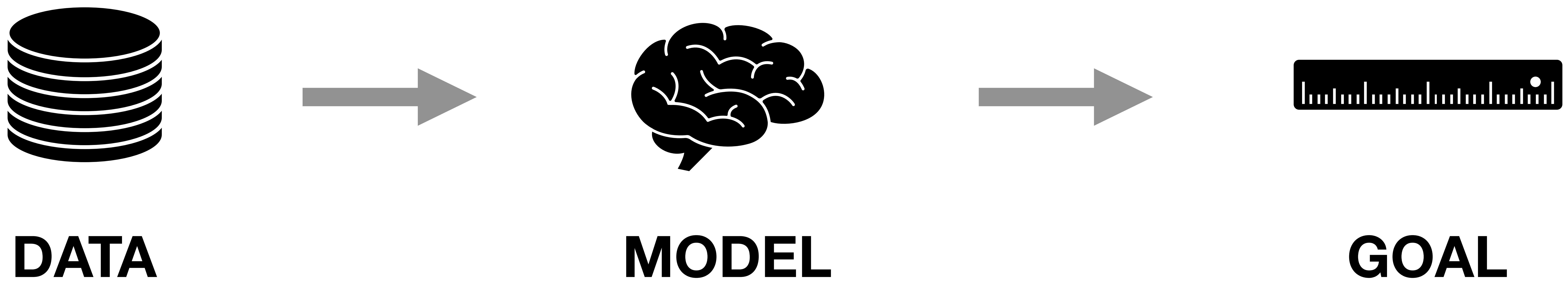
The Invisible Hand of Stability

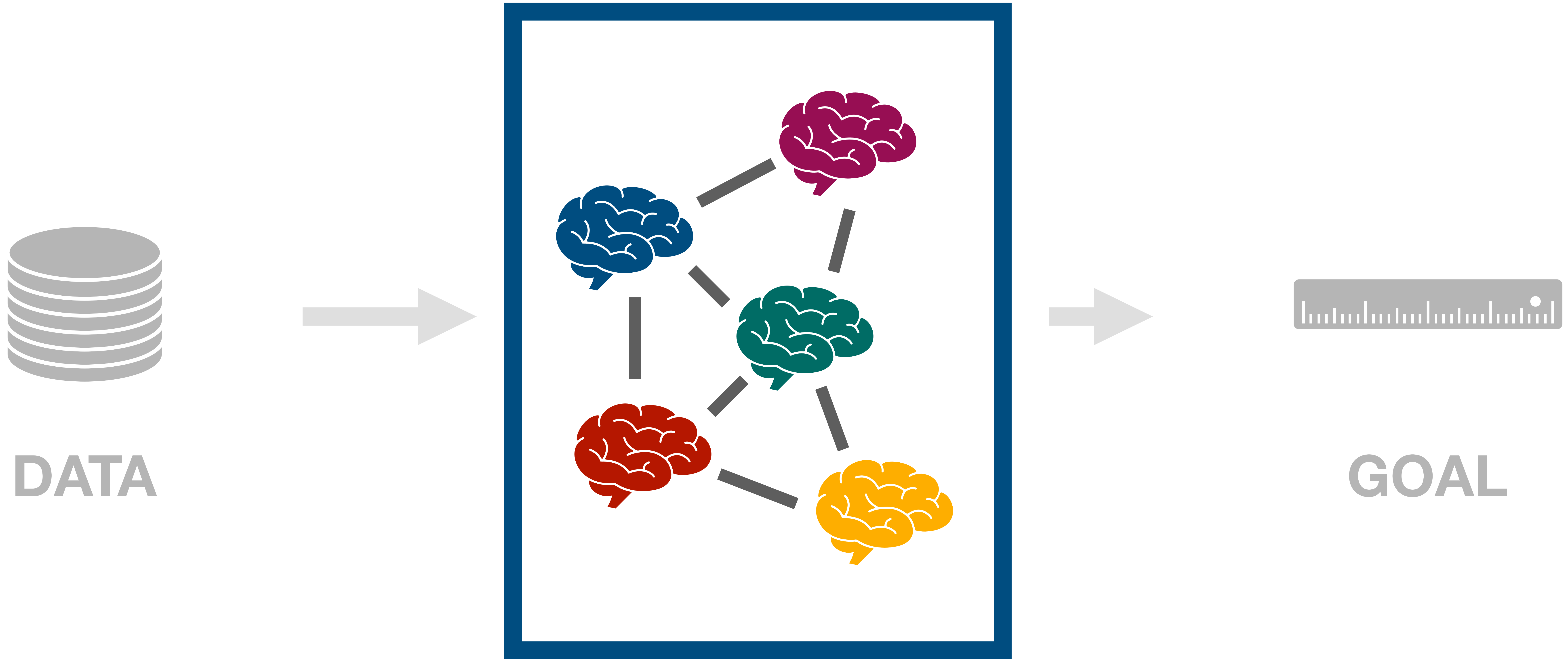
Dynamics of Decentralized Decision Making

Theory Seminar | January 26, 2026

10,000 Foot View of Machine Learning

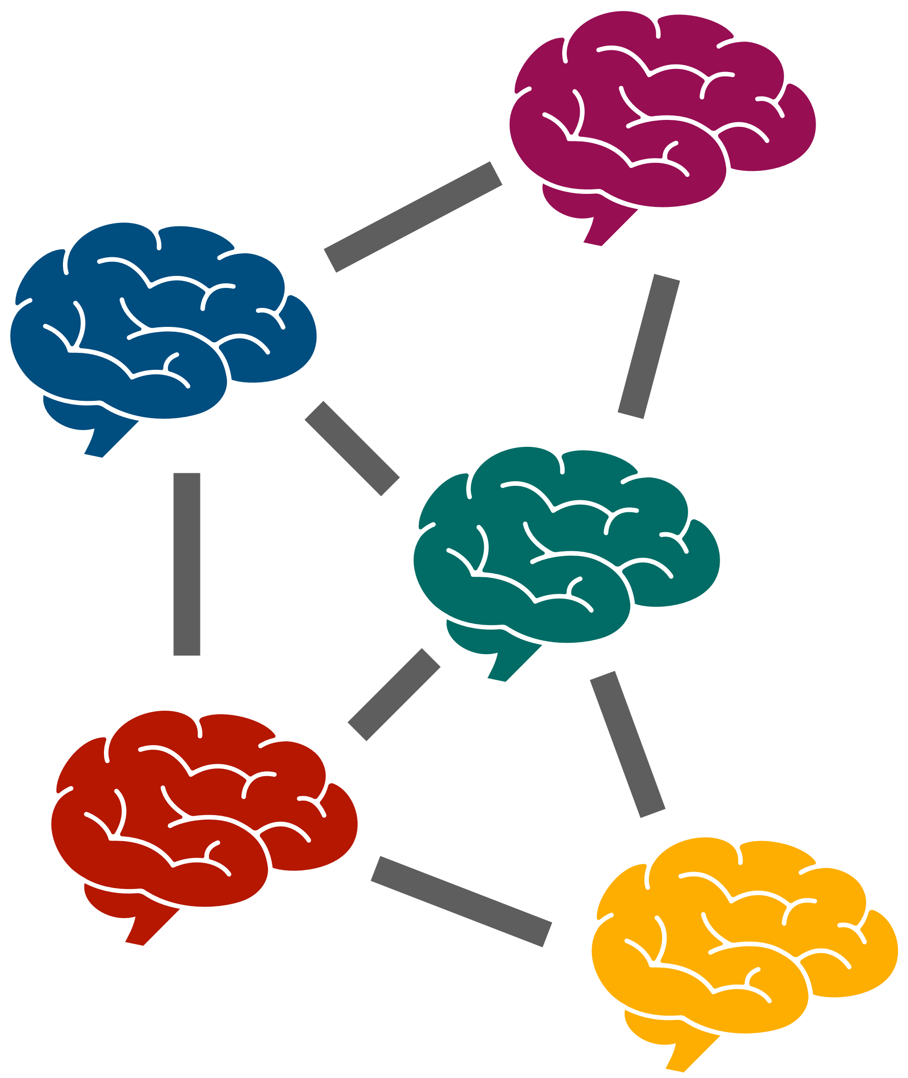


10,000 Foot View of Machine Learning



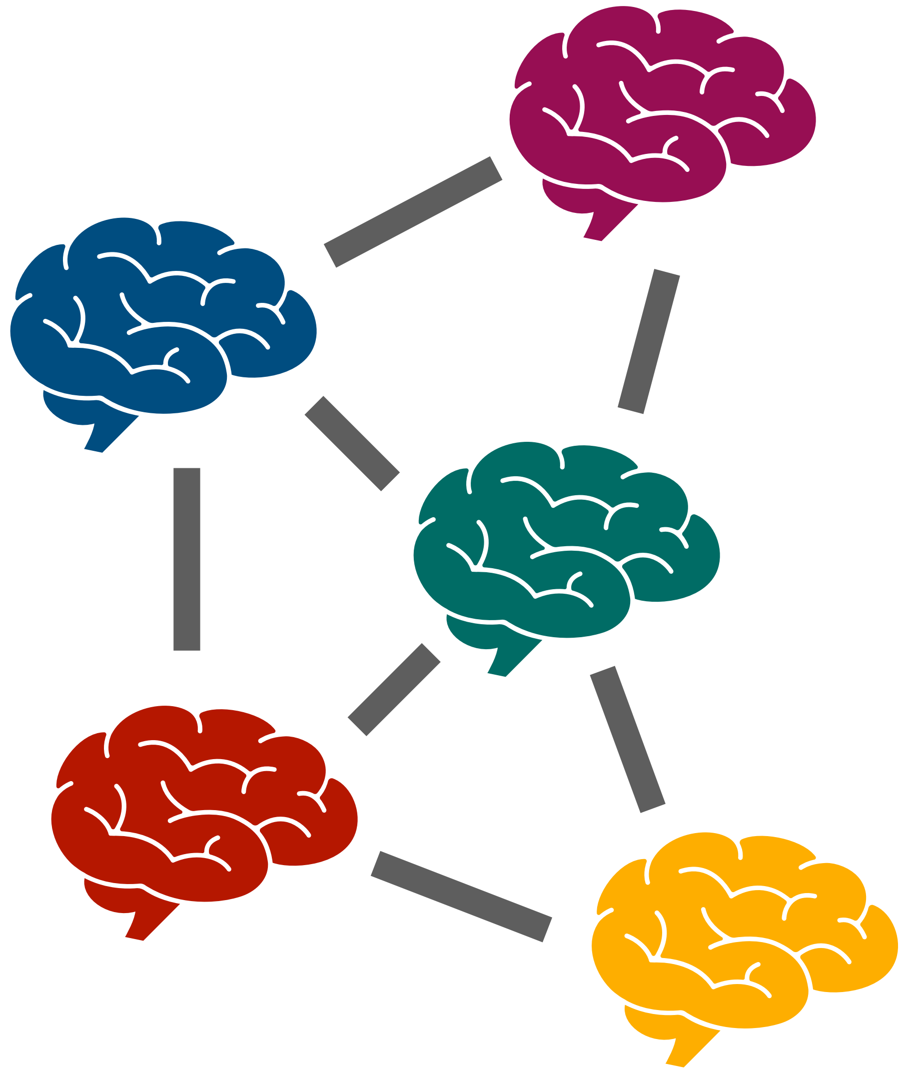
Complex problems often require a division of labor or decentralization.

The Price of Anarchy



Decentralization can introduce constraints on communication and coordination

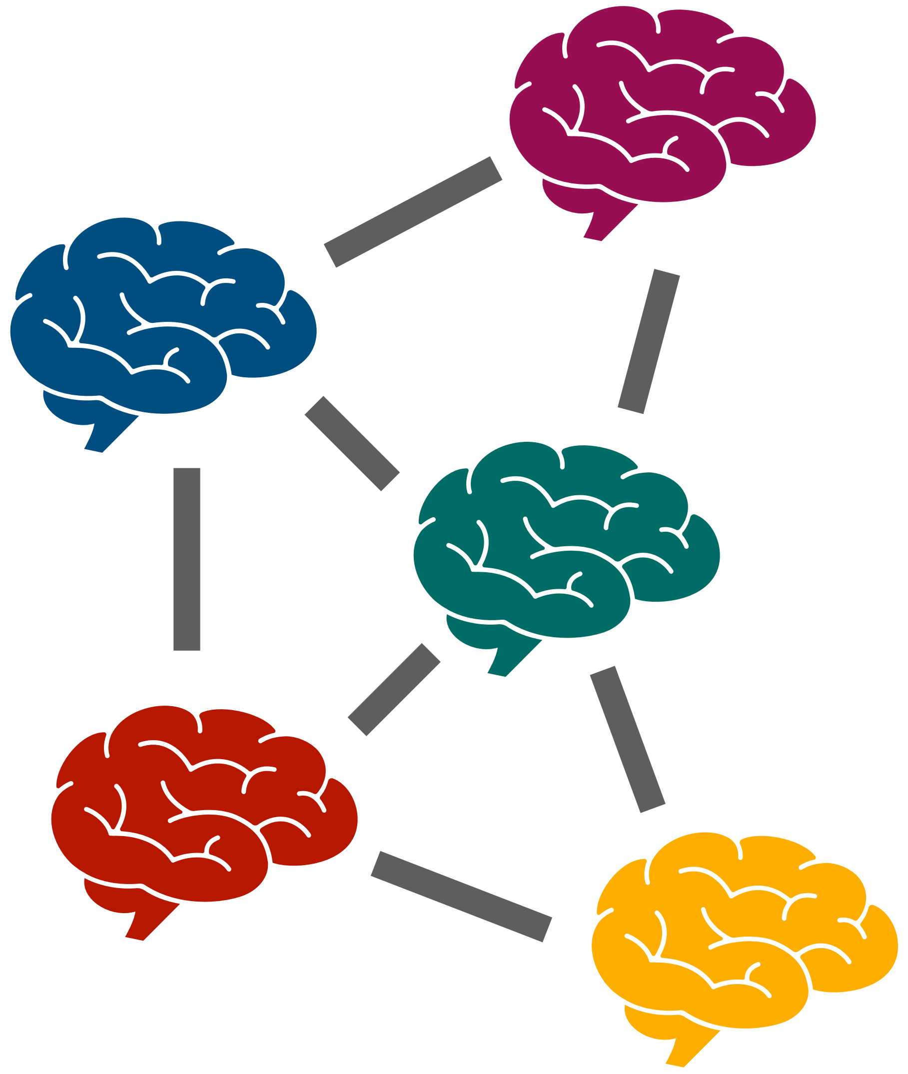
The Price of Anarchy



Decentralization can introduce constraints on communication and coordination

➡ **leading to situations where everyone loses.**

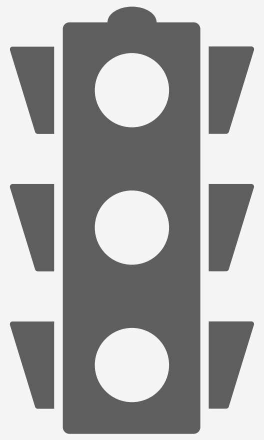
The Price of Anarchy



Decentralization can introduce constraints on communication and coordination

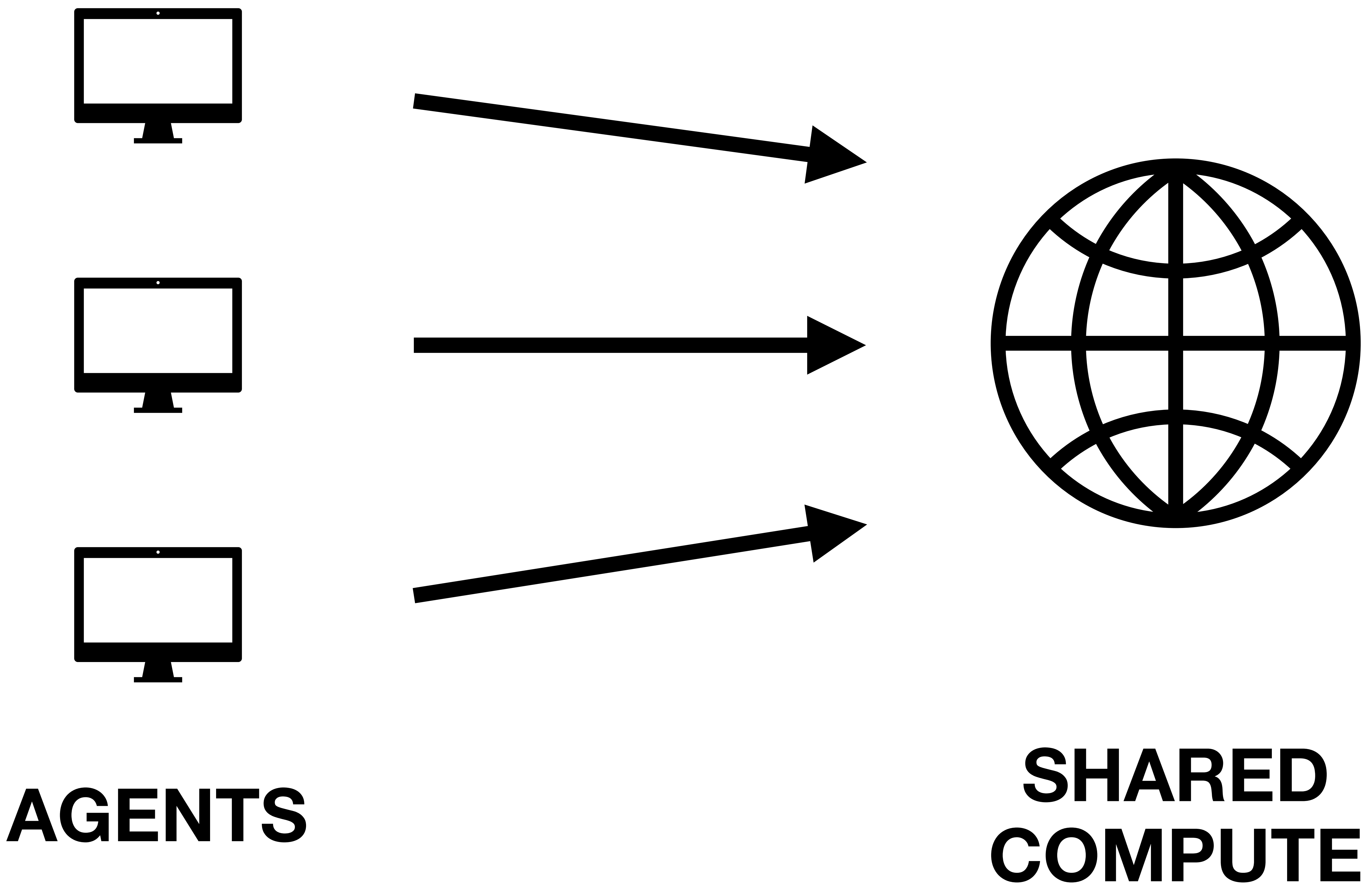
➡ leading to situations where everyone loses.

Prisoner's Dilemma

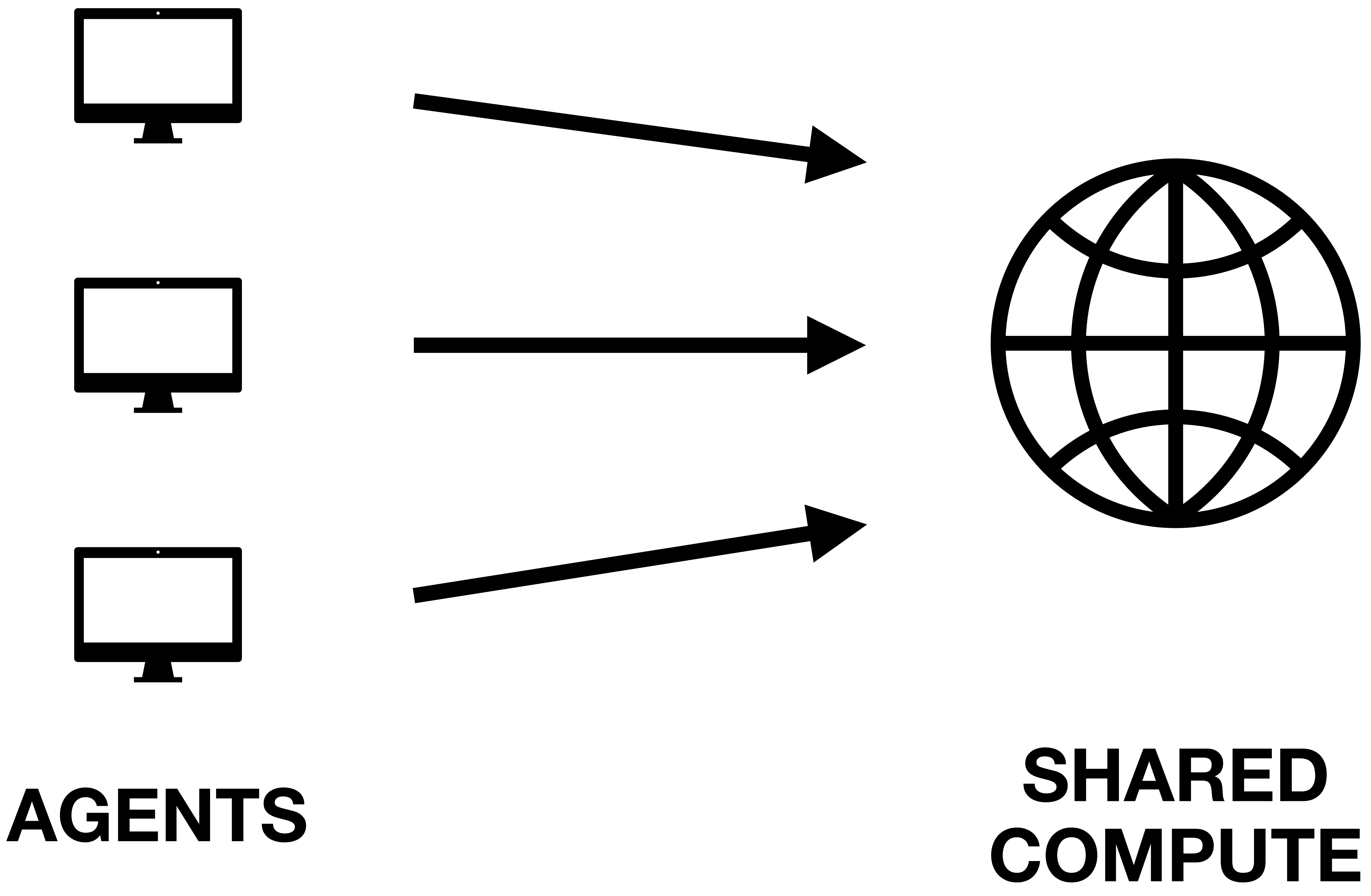


Tragedy of the Commons

Example: Tragedy of the Commons

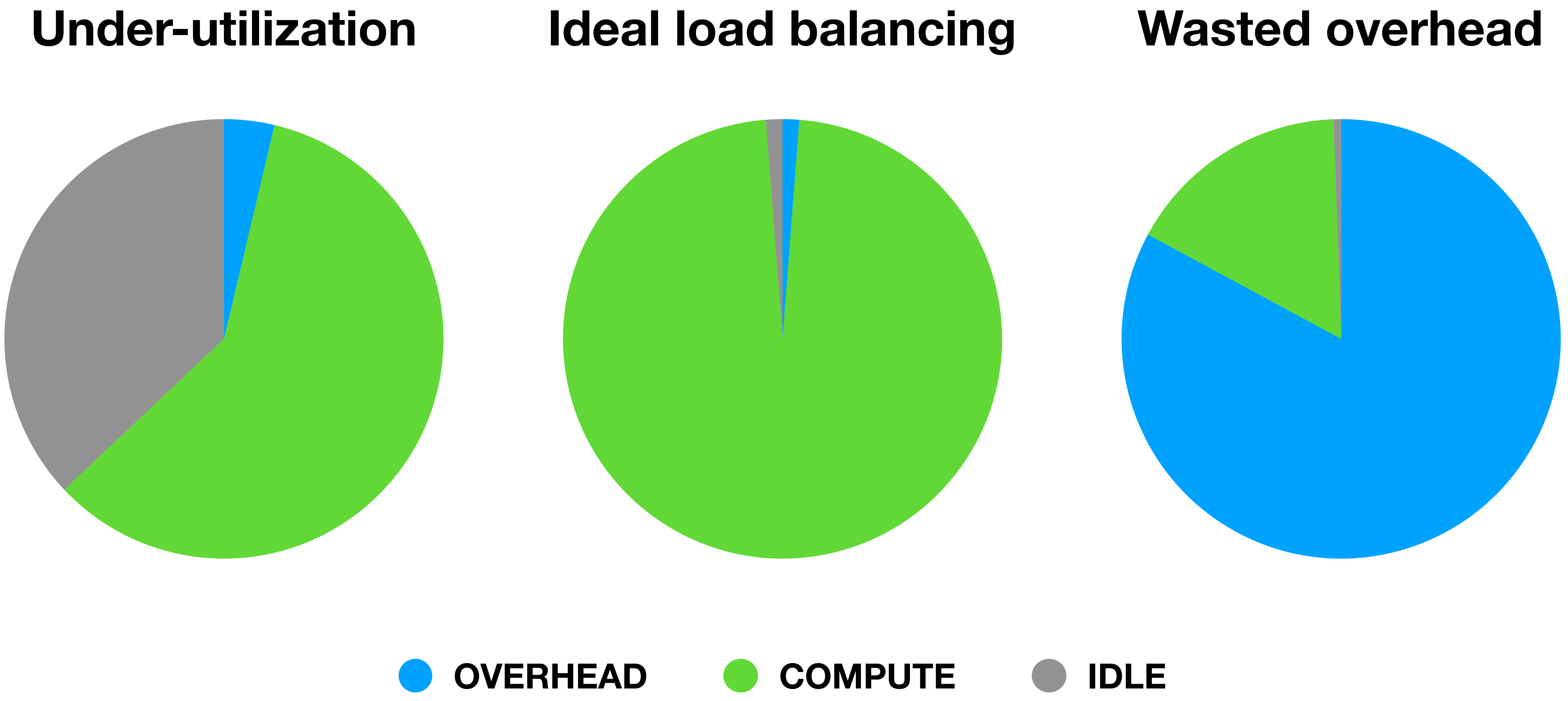


Example: Tragedy of the Commons

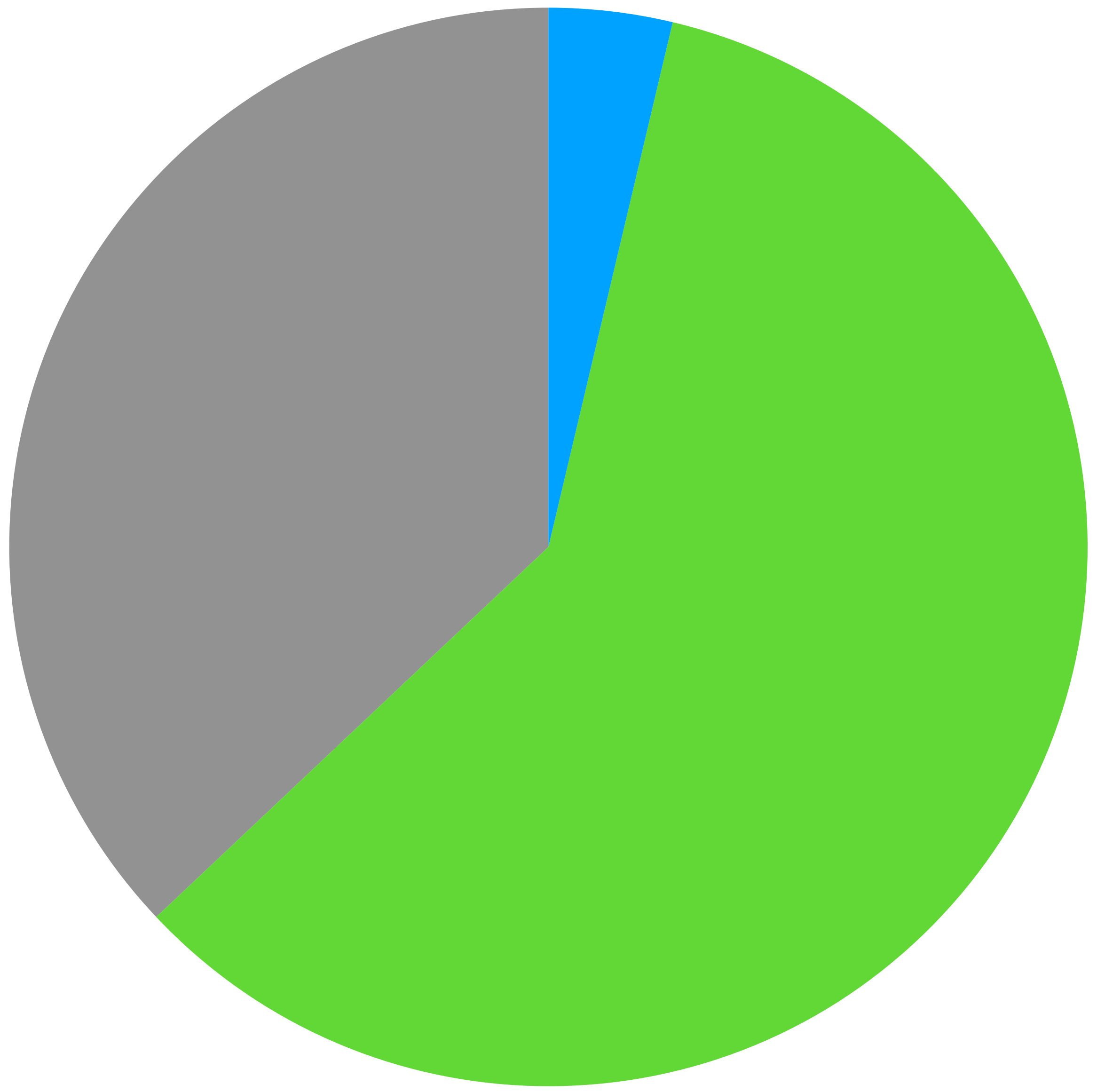


Question: How often should agents send jobs?

Three Possible Worlds



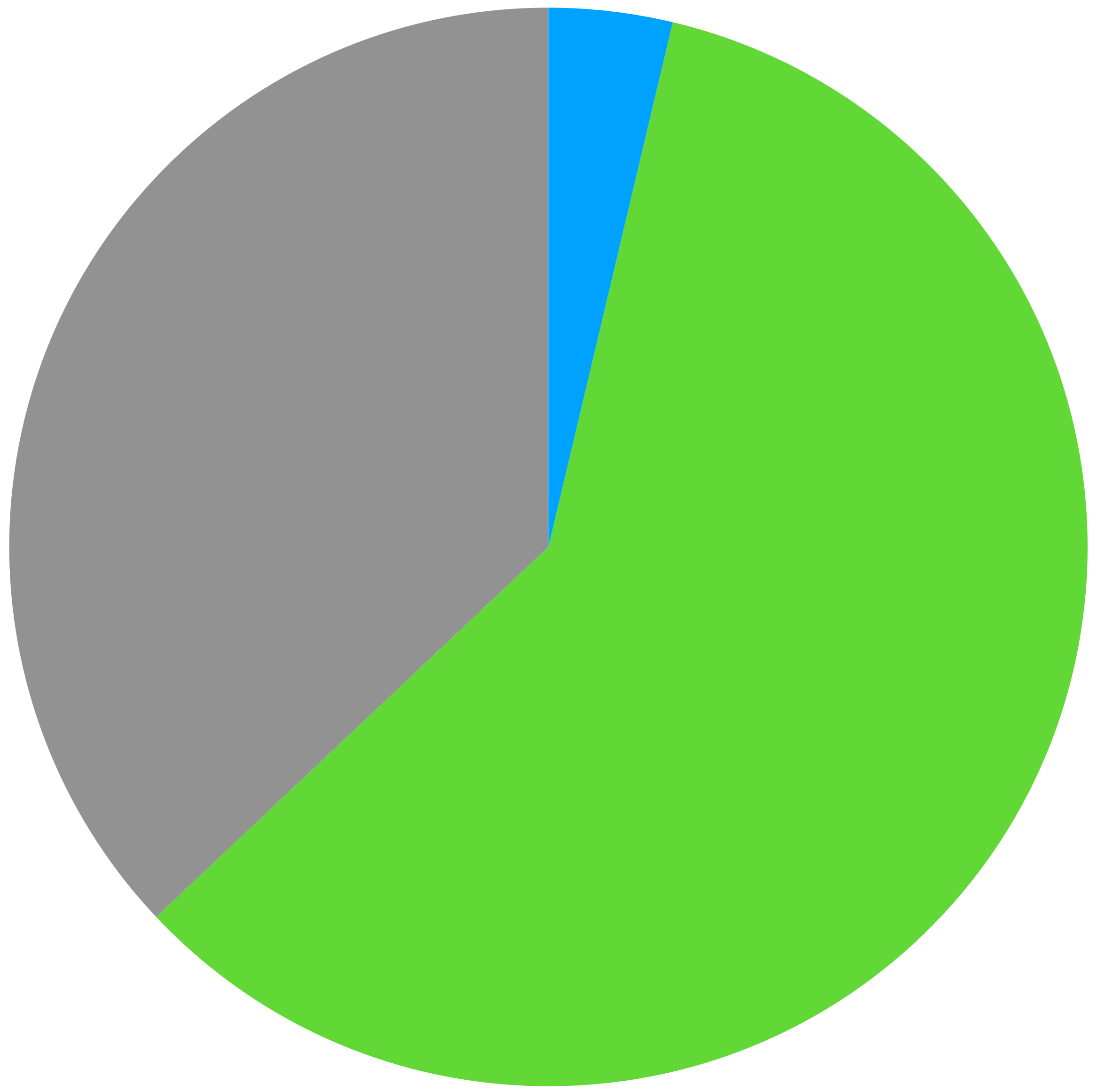
World One: Under-Utilization



Agents send jobs sporadically.

OVERHEAD
COMPUTE
IDLE

World One: Under-Utilization

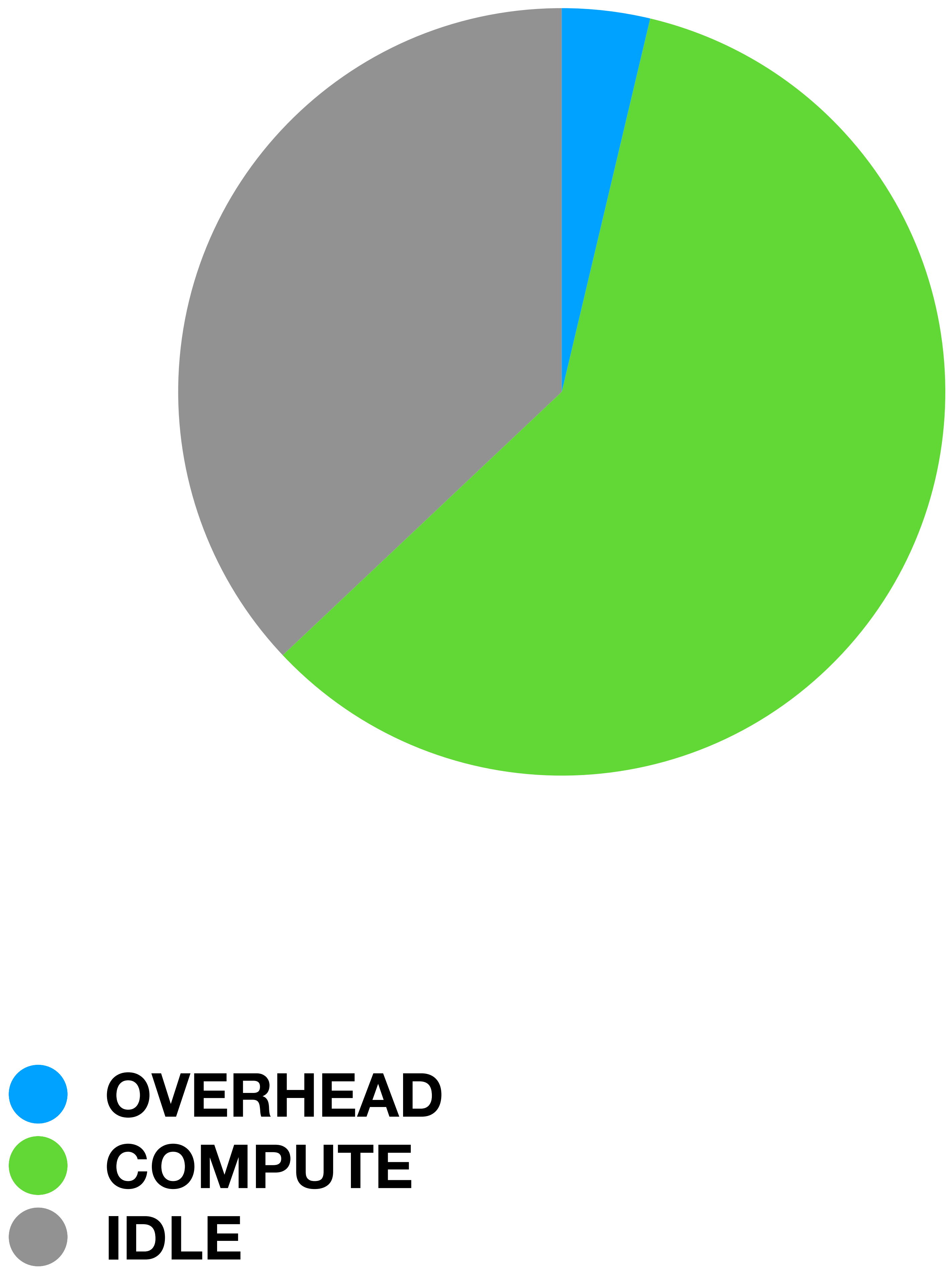


Agents send jobs sporadically.

A lot of idle time: **socially inefficient.**

OVERHEAD
COMPUTE
IDLE

World One: Under-Utilization

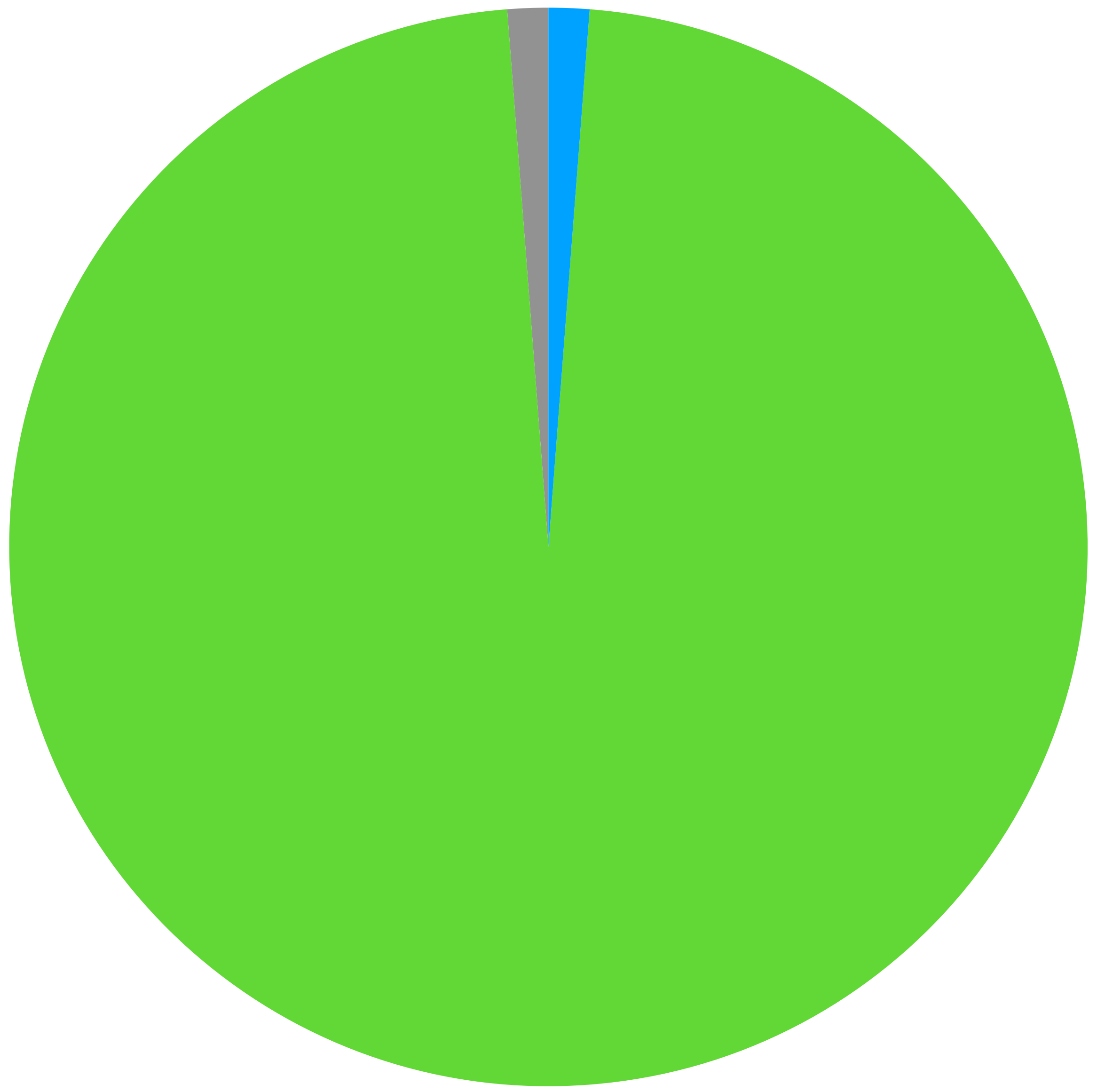


Agents send jobs sporadically.

A lot of idle time: **socially inefficient**.

Everyone benefits from slightly increasing their utilization rate: **unstable**.

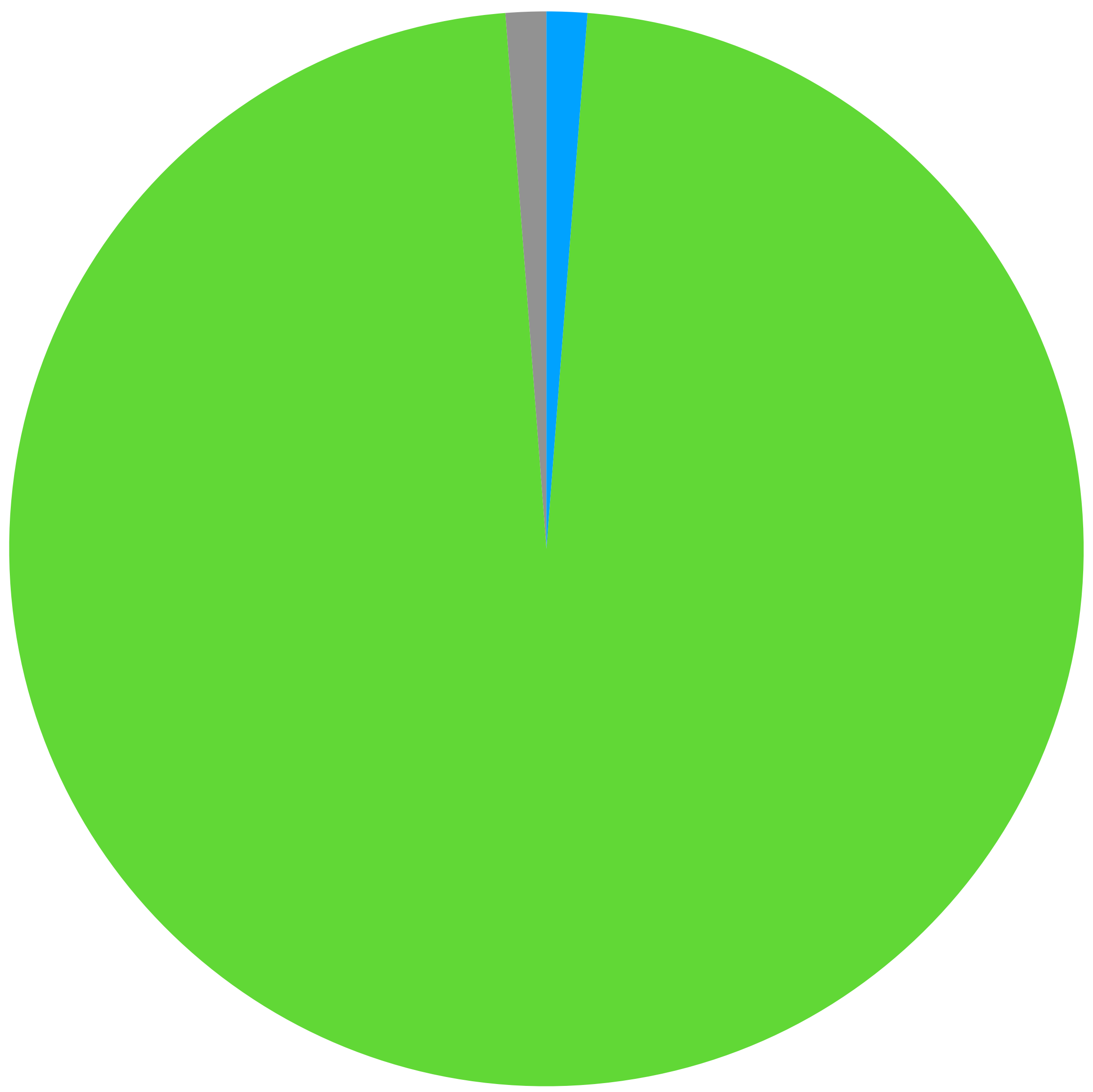
World Two: Perfect Utilization



Demand matches supply.

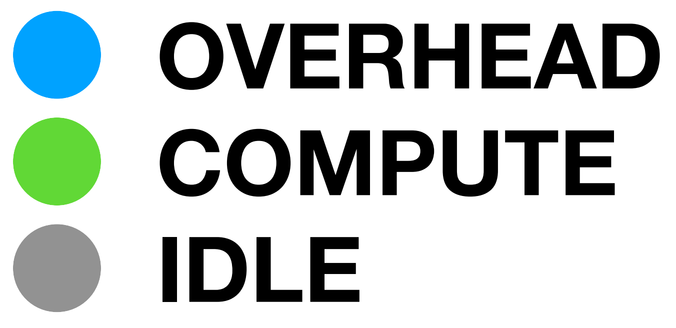
- OVERHEAD
- COMPUTE
- IDLE

World Two: Perfect Utilization



Demand matches supply.

No wasted compute: **socially optimal**.



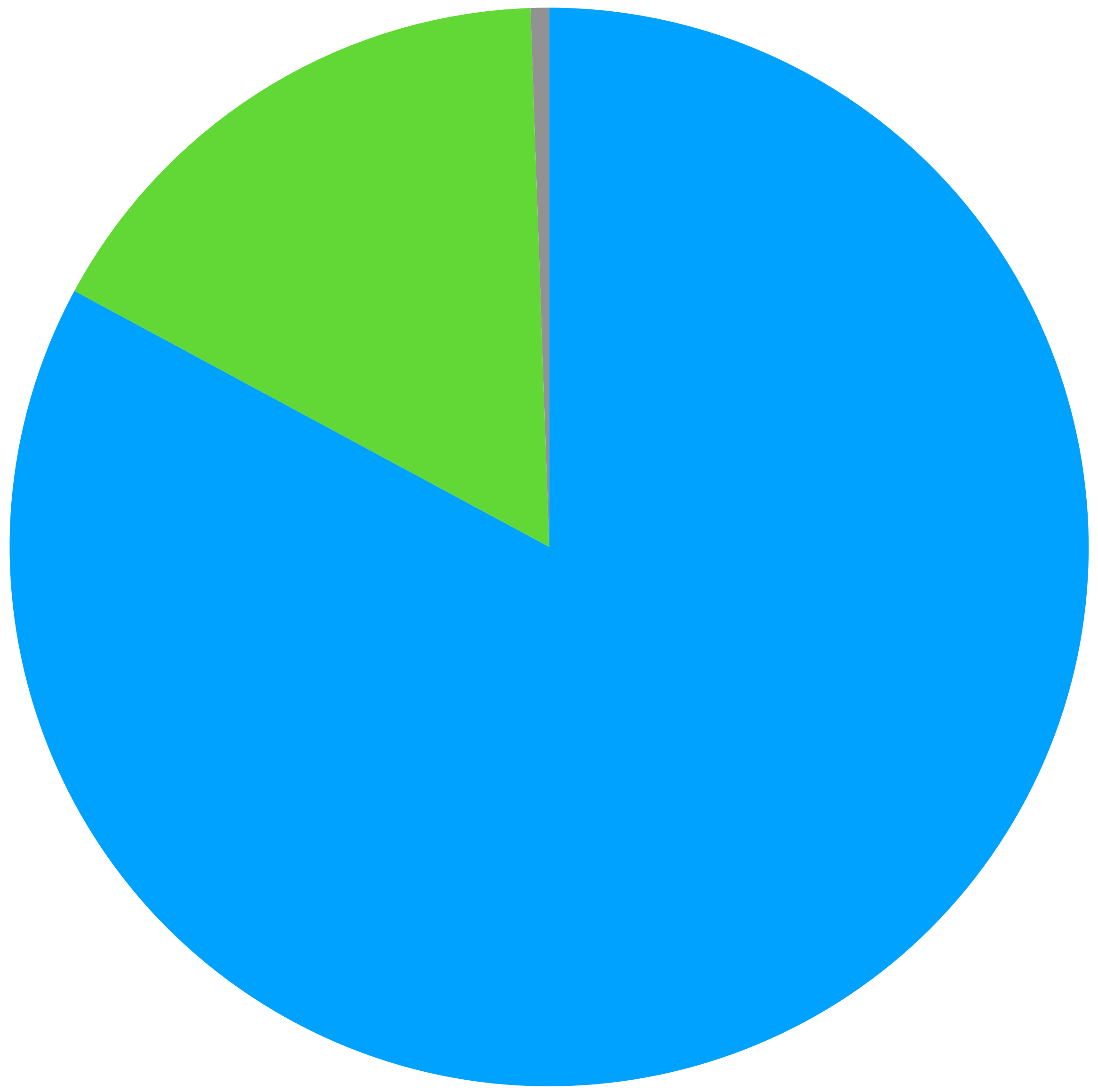
World Two: Perfect Utilization

Demand matches supply.

No wasted compute: **socially optimal**.

Individuals marginally benefit from increasing usage, as long as others do not: **unstable.**

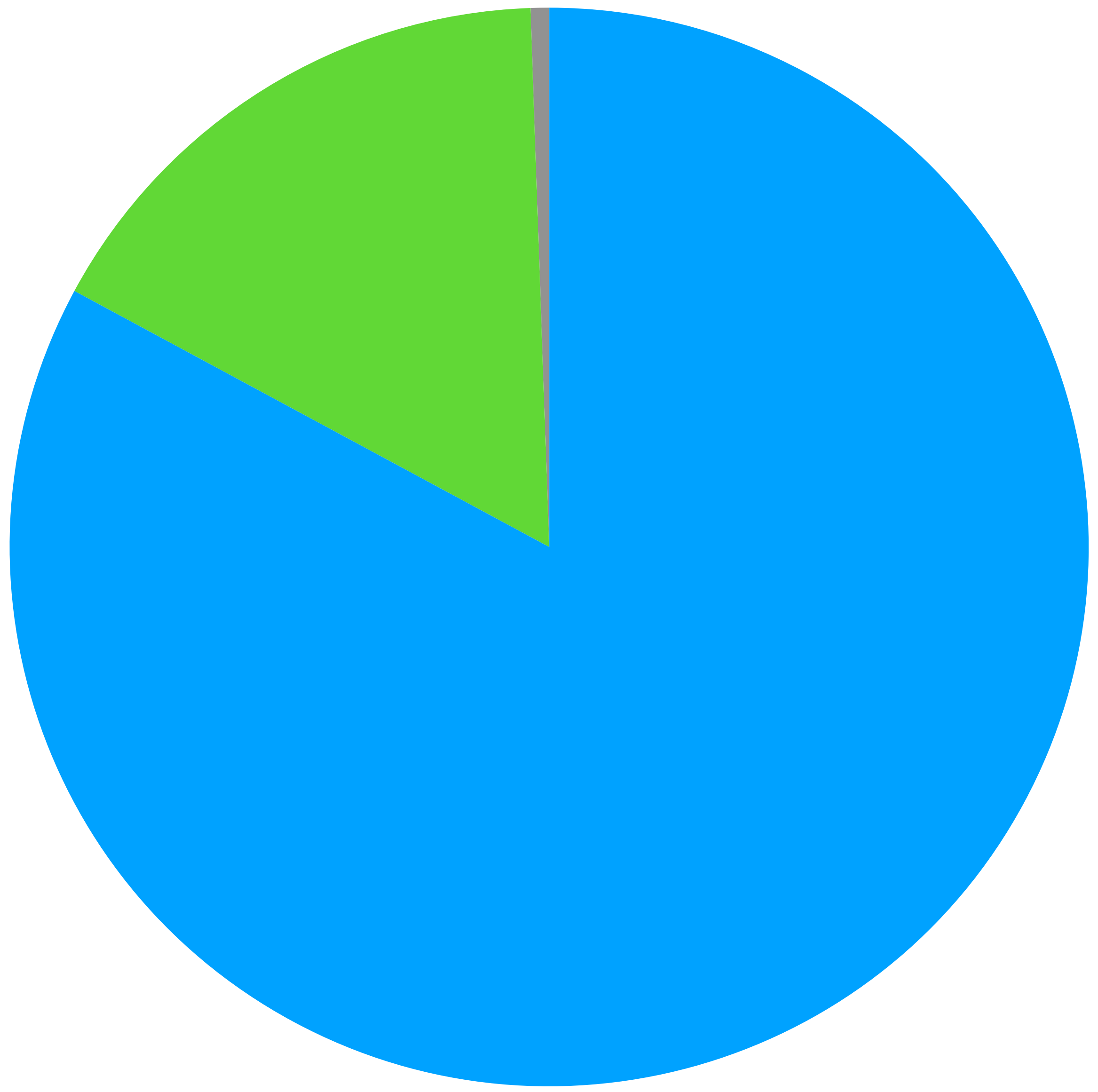
World Three: Over-Utilization



Agents overwhelm the server.

OVERHEAD
COMPUTE
IDLE

World Three: Over-Utilization

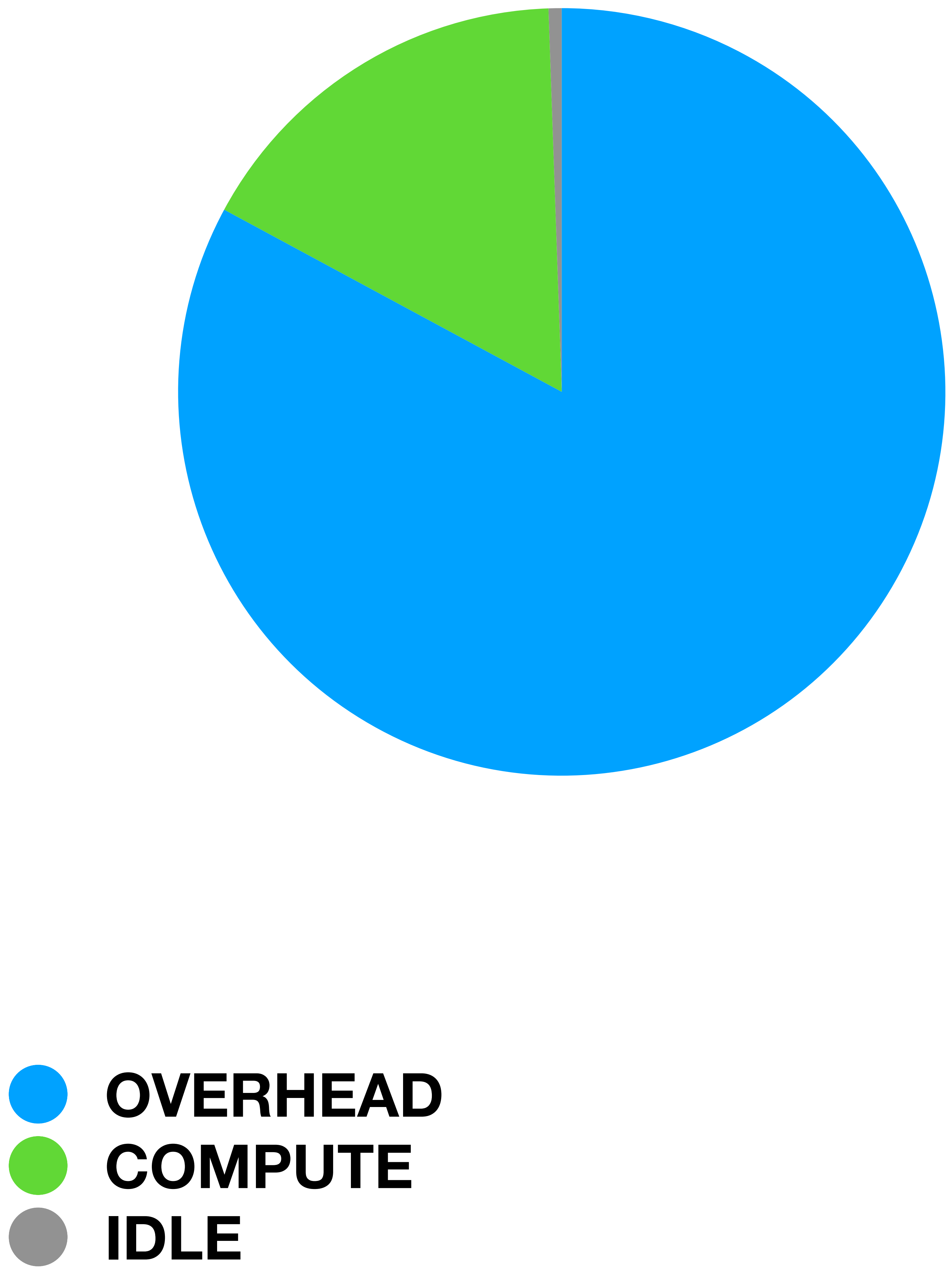


OVERHEAD
COMPUTE
IDLE

Agents overwhelm the server.

Leads to thrashing—most of time is spent in context switching: very **socially inefficient.**

World Three: Over-Utilization

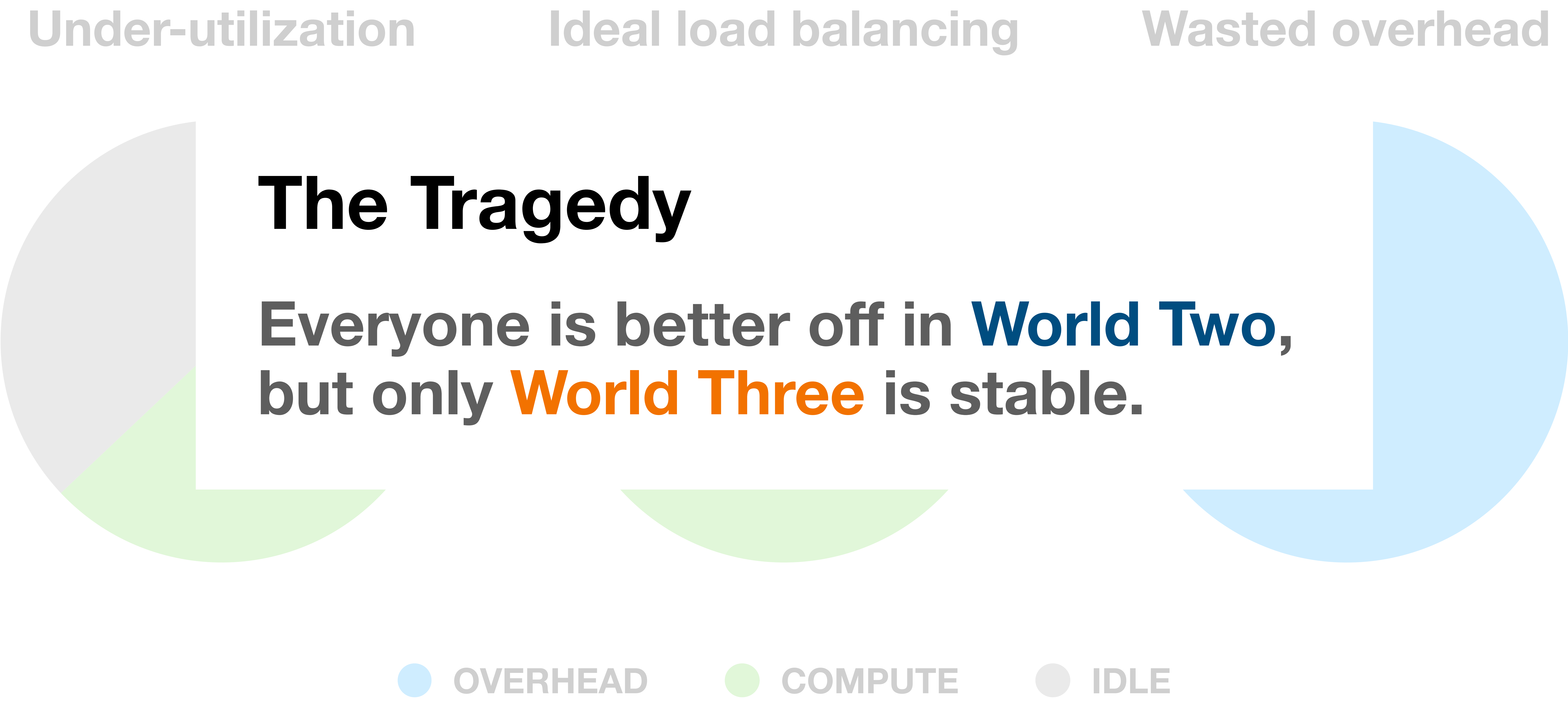


Agents overwhelm the server.

Leads to thrashing—most of time is spent in context switching: very **socially inefficient.**

The individual is marginally worse off by sending fewer jobs: **stable equilibrium**.

Three Possible Worlds



The Tragedy

Everyone is better off in **World Two**, but only **World Three** is stable.

Tragedy of the Commons

Guardrails:

Tragedy of the Commons

Guardrails:

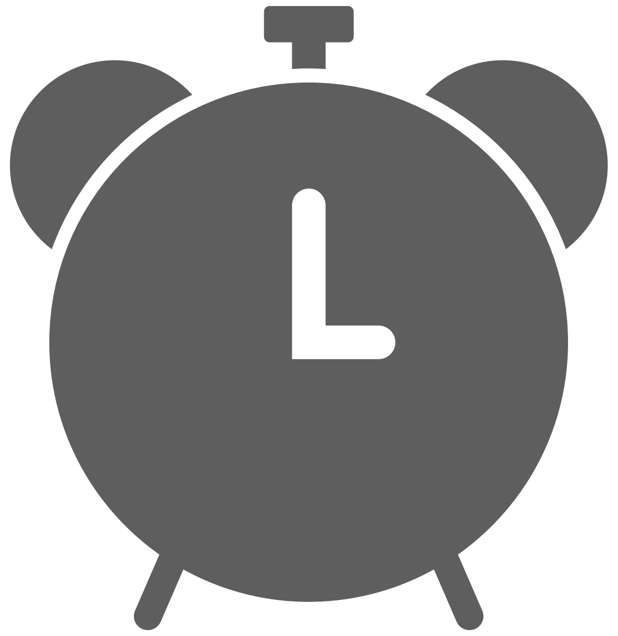
- We can change the **agent behavior** to limit utilization.

Tragedy of the Commons

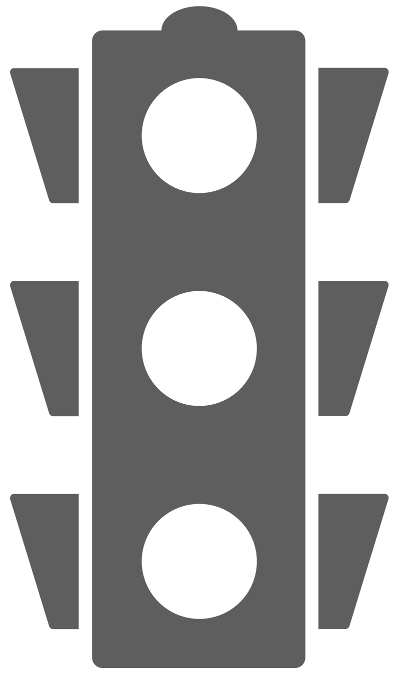
Guardrails:

- We can change the **agent behavior** to limit utilization.
- We can introduce a **central coordinator** to handle usage.

Real-World Solutions

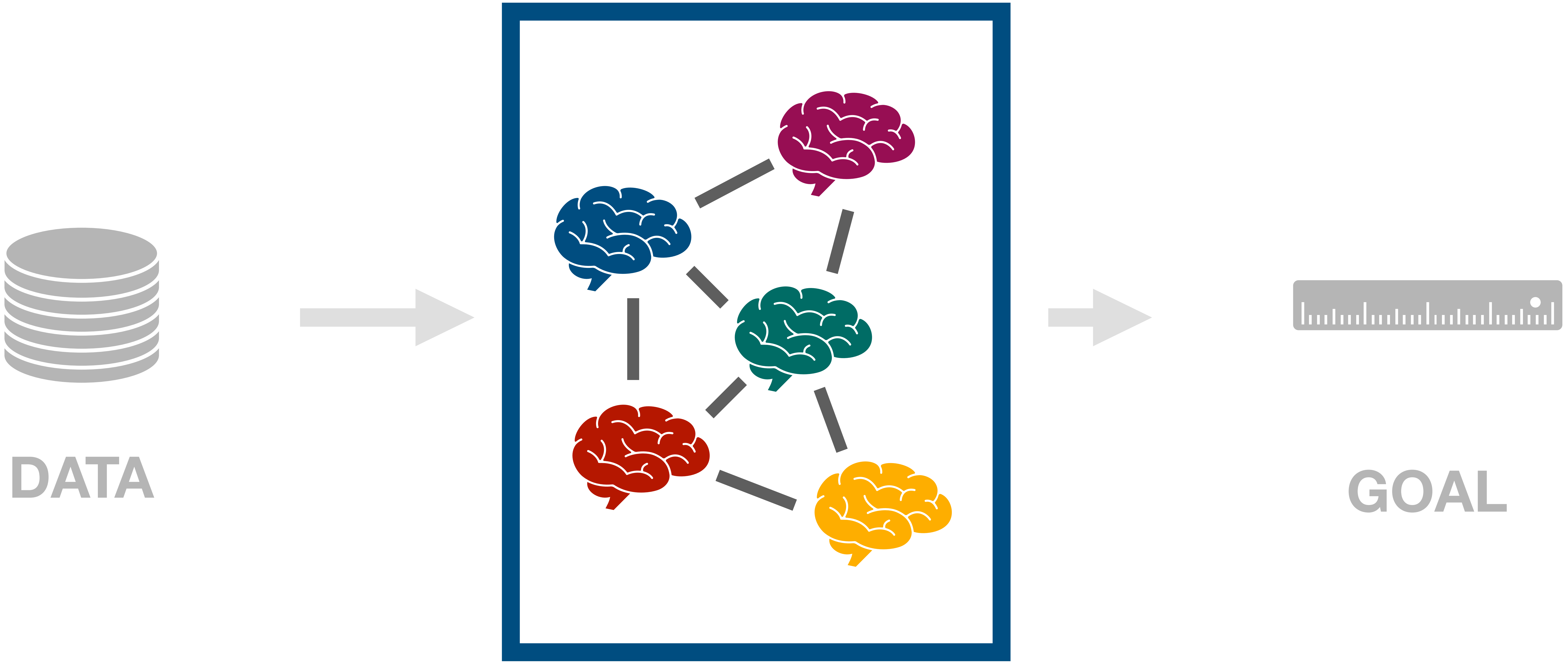


Exponential Backoff
(Client behavior in TCP)



Traffic Light
(Central coordinator)

10,000 Foot View of Machine Learning



But, what happens if we cannot *explicitly* design the agent behaviors?
In modern ML systems, the behaviors of agents are often learned.

Nash Equilibria

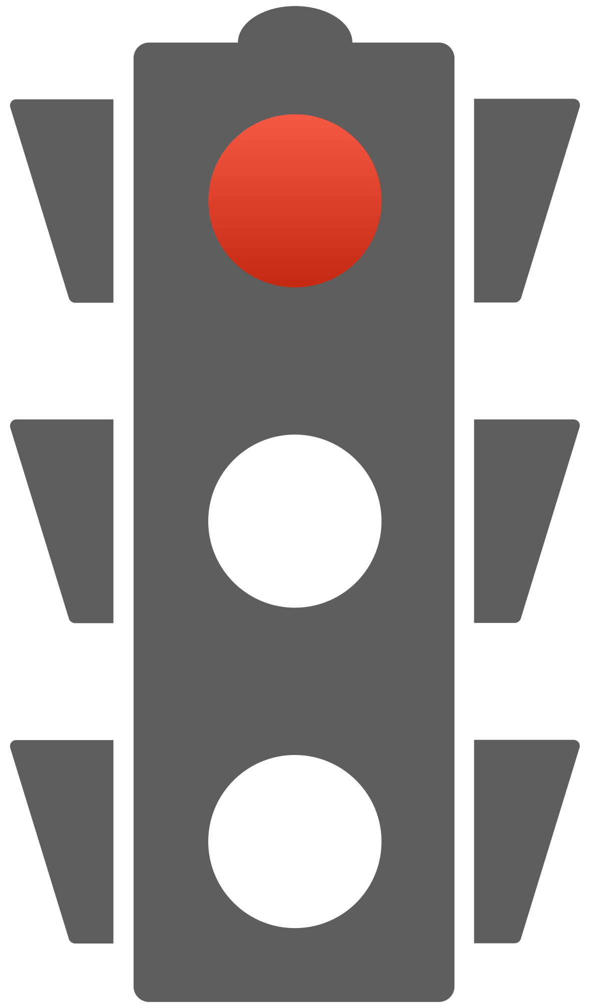
Nash equilibria (NE) describe behaviors of multi-agent systems that may have **long-term relevance**.

Nash Equilibria

Nash equilibria (NE) describe behaviors of multi-agent systems that may have **long-term relevance**.

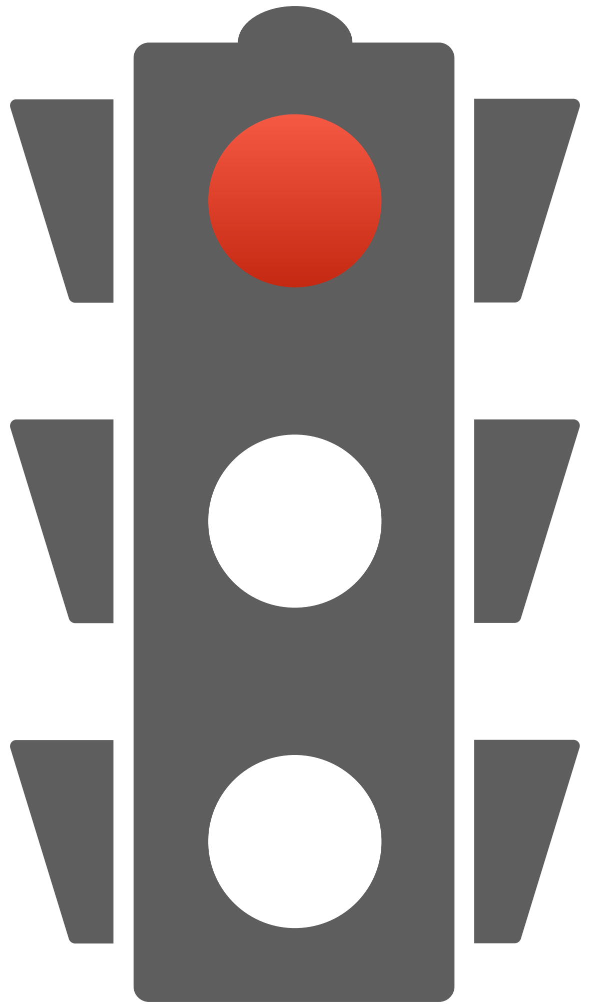
But not all NE are <u>stable</u>, and not all are <u>good</u>.

Prior Work: Stability of Social Conventions



Dynamics of multi-agent systems around strict Nash equilibria are well-understood.

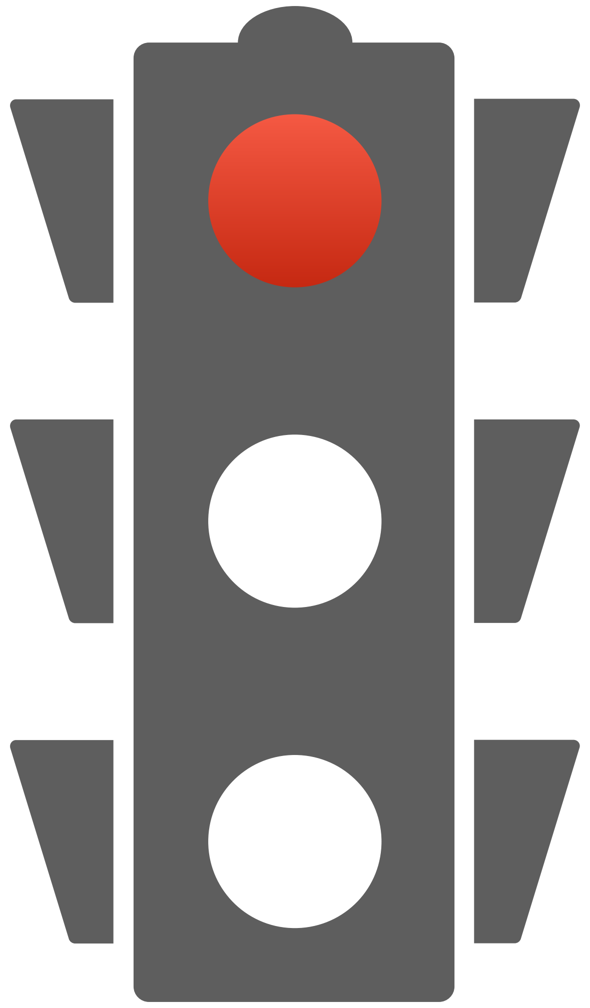
Prior Work: Stability of Social Conventions



Dynamics of multi-agent systems around strict Nash equilibria are well-understood.

Example. Once enough people agree **RED/GREEN** means **STOP/GO**, everyone else is forced to adopt this convention.

Prior Work: Stability of Social Conventions



Dynamics of multi-agent systems around strict Nash equilibria are well-understood.

Example. Once enough people agree **RED/GREEN** means **STOP/GO**, everyone else is forced to adopt this convention.

Stability does not guarantee collective rationality.

Stability in Non-Strict Equilibria

What about dynamics around
non-strict Nash equilibria?

Stability in Non-Strict Equilibria

What about dynamics around non-strict Nash equilibria?

Example. When the job market is at equilibrium, many career options are viable, leading to a mixed population.

Stability in Non-Strict Equilibria

What about dynamics around non-strict Nash equilibria?

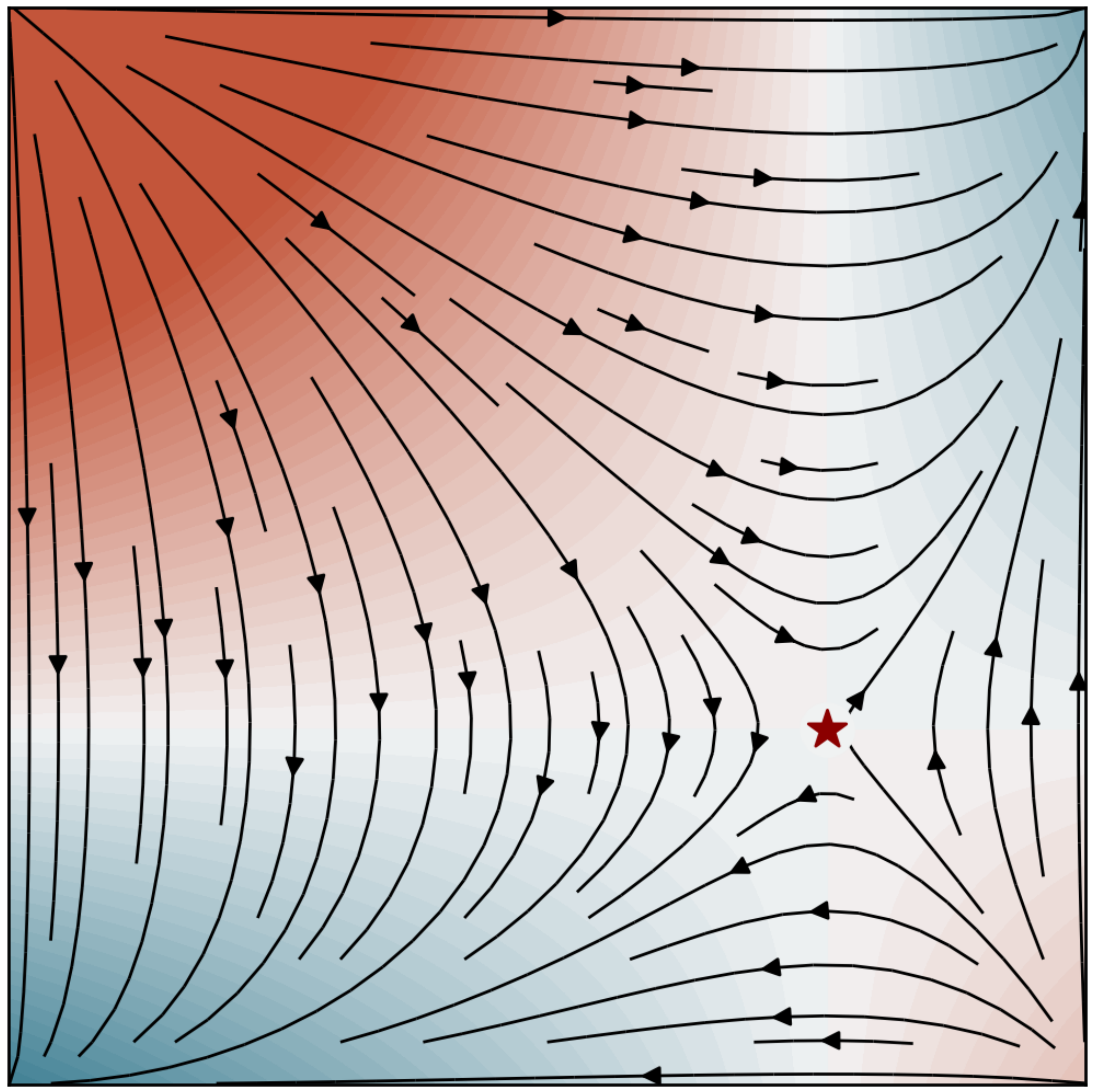
Example. When the job market is at equilibrium, many career options are viable, leading to a mixed population.

Stability guarantees collective rationality.

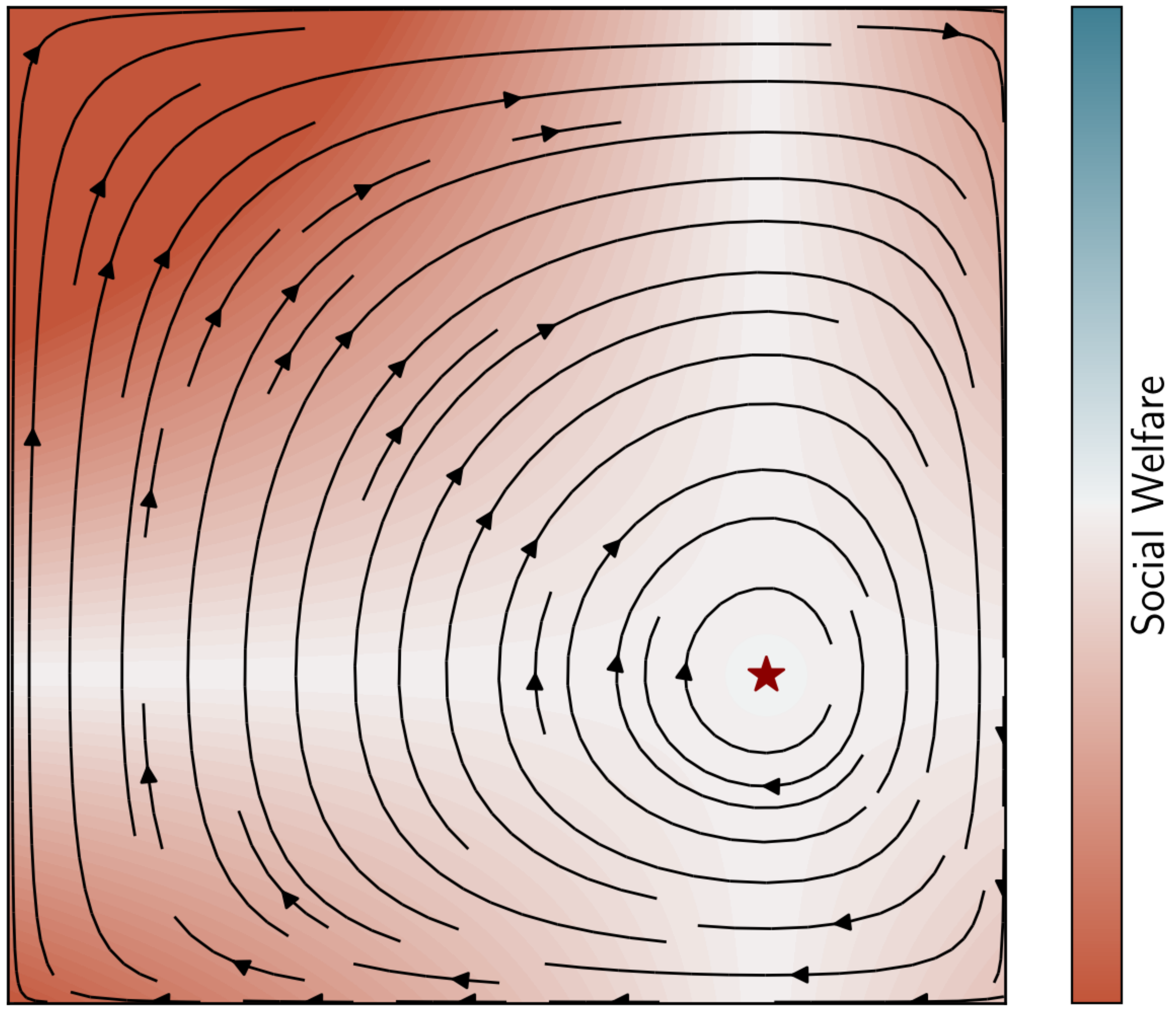
Learnable Mixed Nash Equilibria are Collectively Rational

Geelon So and Yi-An Ma, Preprint 2025

Comparison of Dynamics Around Non-Strict NE



Unstable Nash equilibria are not strategically Pareto optimal.



Uniformly stable Nash equilibria are strategically Pareto optimal.

Takeaways

- **Modern ML often implements multi-agent solutions.**
- **Decentralization introduces structural constraints.**
- **What are the ramifications and when are guardrails needed?**

Contents

I. A crash course in game theory

Hallmark of a Game

Decisions are made **individually**, but the **outcome** depends on **collective** choices.

Game Components

- Multiple **decision makers** (players, agents, learners)
- Possibly distinct **goals** (utilities, tasks, objectives)
- Disjoint **decision variables** (strategies, policies, parameters)

Optimization v. Games

Single-Objective Optimization	Multi-Objective Optimization	Games
One objective	Many objectives	Many objectives
One decision maker	One decision maker	Many decision makers
	Shared parameters	Disjoint parameters

Mathematical Formalism

An *N*-player game:

- **Players** indexed by $n \in \{1, \ldots, N\}$

Mathematical Formalism

An *N*-player game:

- **Players** indexed by $n \in \{1, \ldots, N\}$
- Each player has a **decision variable** $x_n \in \Omega_n$

Mathematical Formalism

An *N*-player game:

- **Players** indexed by $n \in \{1, \ldots, N\}$
- Each player has a **decision variable** $x_n \in \Omega_n$
    - Joint decision: $\mathbf{x} = (x_1, \ldots, x_N) \in \boldsymbol{\Omega}$

$\boldsymbol{\Omega} := \Omega_1 \times \cdots \times \Omega_N$

Mathematical Formalism

An *N*-player game:

- **Players** indexed by $n \in \{1,\ldots,N\}$
- Each player has a **decision variable** $x_n \in \Omega_n$
  - Joint decision: $\mathbf{x} = (x_1, \ldots, x_N) \in \boldsymbol{\Omega}$
  - Notation: $\mathbf{x}_{-n} := (x_1, \ldots, x_{n-1}, x_{n+1}, \ldots, x_N)$

Mathematical Formalism

An *N*-player game:

- **Players** indexed by $n \in \{1, \ldots, N\}$
- Each player has a **decision variable** $x_n \in \Omega_n$
  - Joint decision: $\mathbf{x} = (x_1, \ldots, x_N) \in \mathbf{\Omega}$
  - Notation: $\mathbf{x}_{-n} := (x_1, \ldots, x_{n-1}, x_{n+1}, \ldots, x_N)$
- Each player wants to maximize their **utility** $u_n : \mathbf{\Omega} \to \mathbb{R}$

$$u_n(\mathbf{x}) \equiv u_n(x_n; \mathbf{x}_{-n}).$$

Nash Equilibrium

Definition. A joint strategy $\mathbf{x}^\star$ is a **Nash equilibrium** if no player can unilaterally improve their own utility

Nash Equilibrium

Definition. A joint strategy $\mathbf{x}^\star$ is a **Nash equilibrium** if no player can unilaterally improve their own utility

$$u_n(\mathbf{x}^\star) = \max_{x_n \in \Omega_n} u_n(x_n; \mathbf{x}^\star_{-n}).$$

Nash Equilibrium

Definition. A joint strategy $\mathbf{x}^\star$ is a **Nash equilibrium** if no player can unilaterally improve their own utility

$$u_n(\mathbf{x}^\star) = \max_{x_n \in \Omega_n} u_n(x_n; \mathbf{x}^\star_{-n}).$$

This is a notion of **individual rationality**: players would need to *collaborate* to further improve their utilities.

Pareto Optimum

Definition. A joint strategy $\mathbf{x}^\star$ is (weakly) **Pareto optimal** if there is no way to improve all players at the same time

Pareto Optimum

Definition. A joint strategy $\mathbf{x}^\star$ is (weakly) Pareto optimal if there is no way to improve all players at the same time

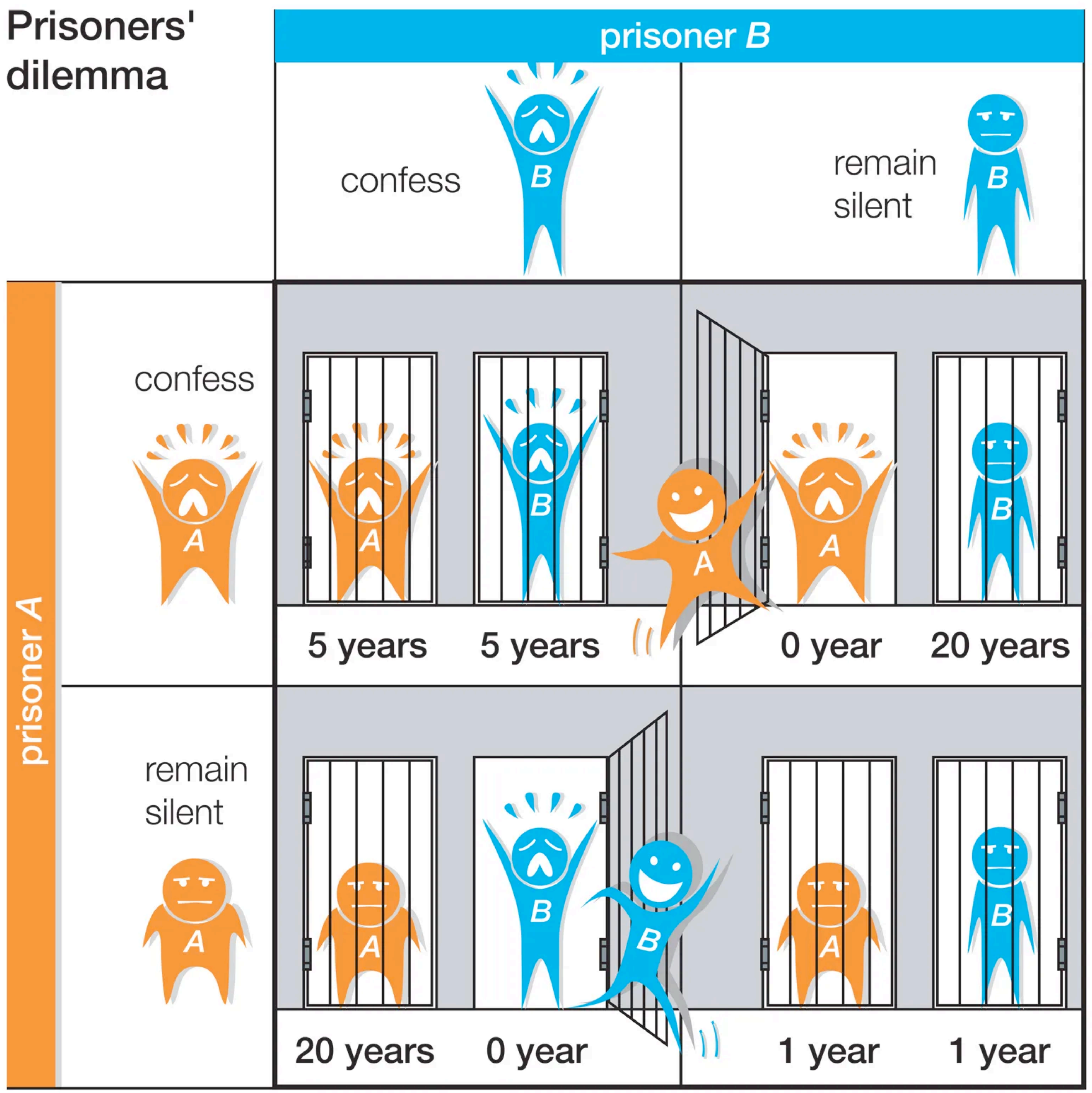
$$\nexists \mathbf{x} \in \boldsymbol{\Omega} \quad \text{s.t.} \quad u_n(\mathbf{x}) > u_n(\mathbf{x}^\star) \text{ for all } n \in [N].$$

Pareto Optimum

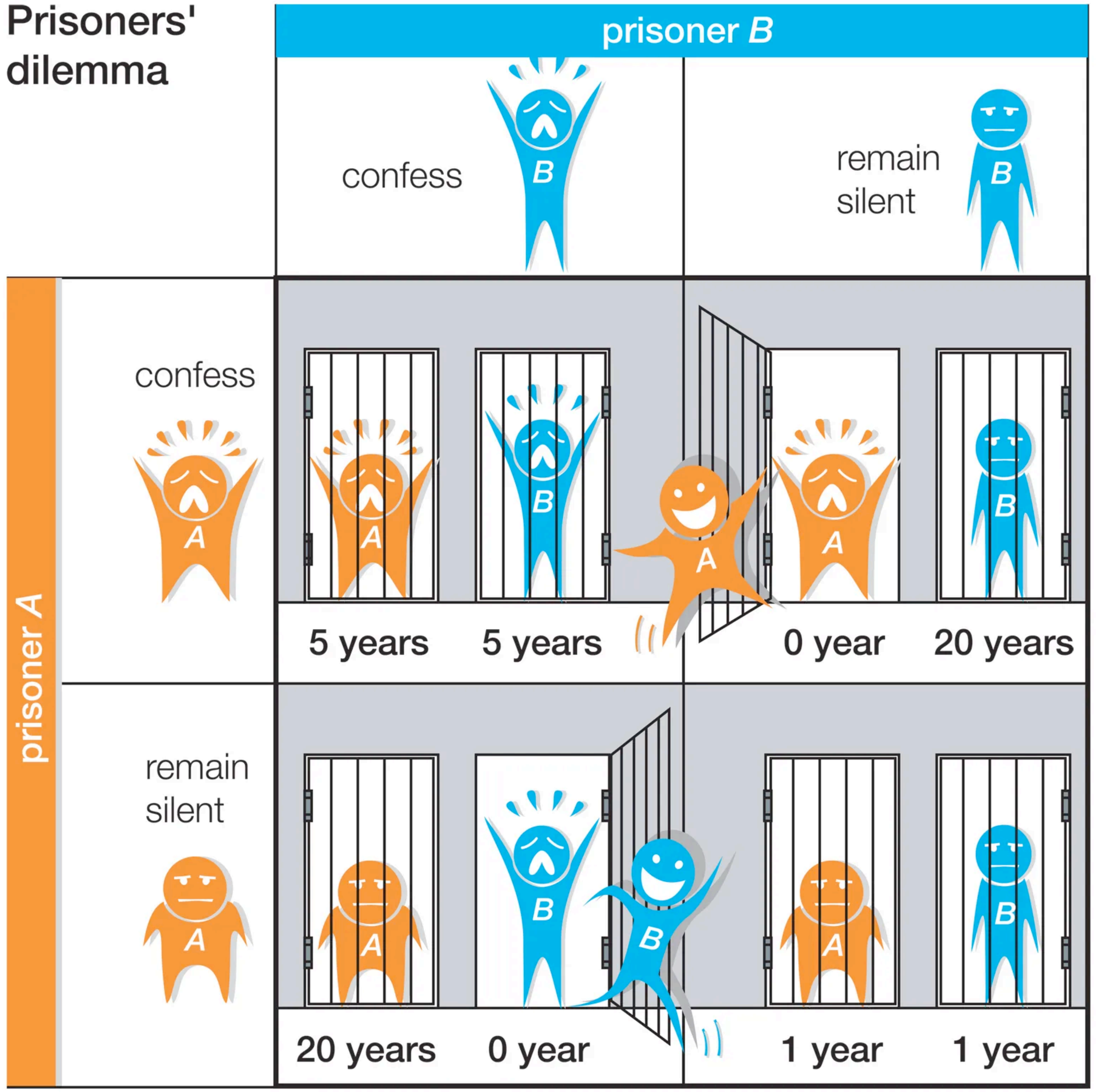
Definition. A joint strategy $\mathbf{x}^\star$ is (weakly) **Pareto optimal** if there is no way to improve all players at the same time

$$\nexists \mathbf{x} \in \boldsymbol{\Omega} \quad \text{s.t.} \quad u_n(\mathbf{x}) > u_n(\mathbf{x}^\star) \text{ for all } n \in [N].$$

It would be **collectively irrational** to jointly play $\mathbf{x}^\star$ when everyone is happier with $\mathbf{x}$.

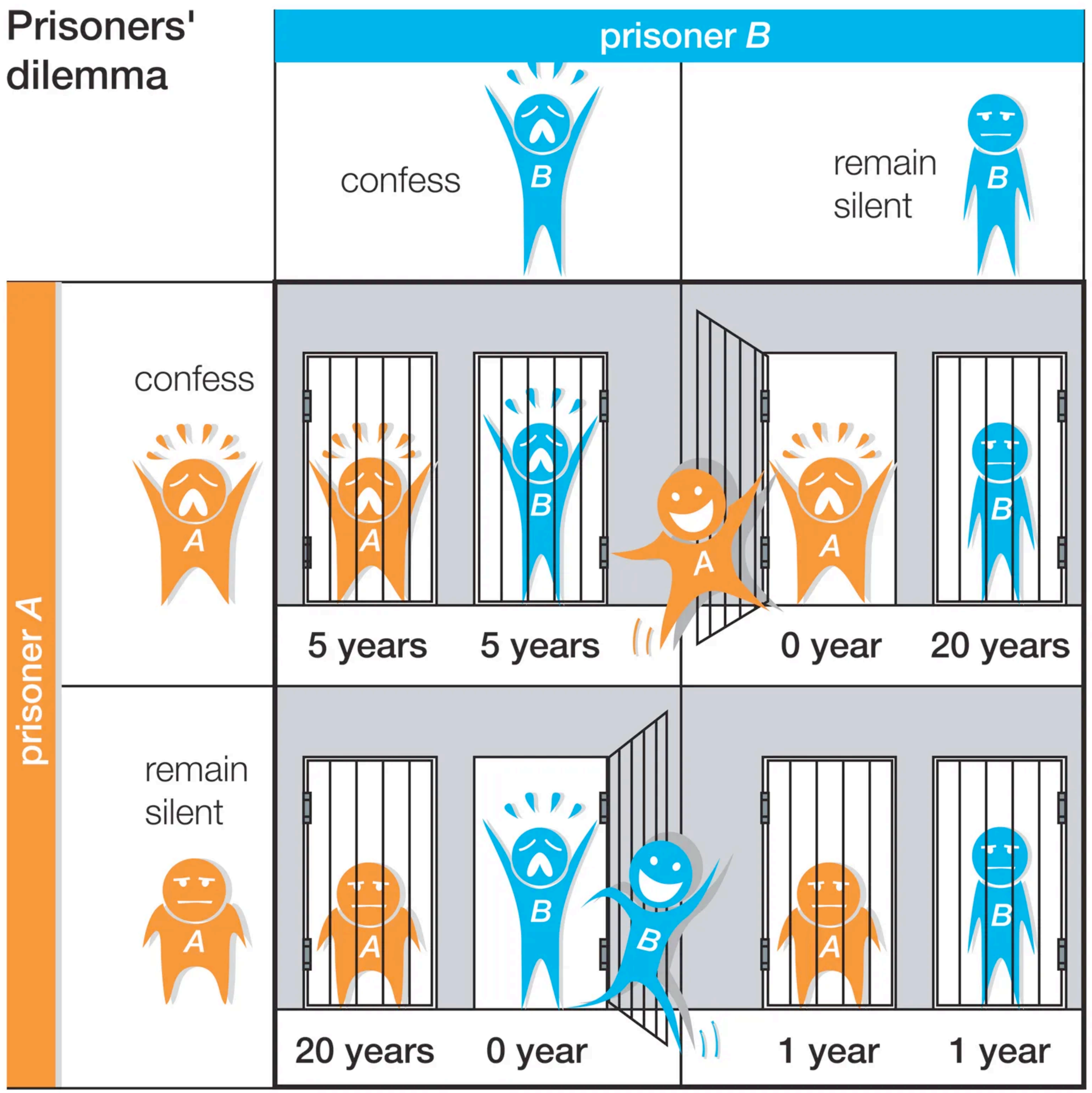


Nash equilibria do not need to be Pareto optimal.



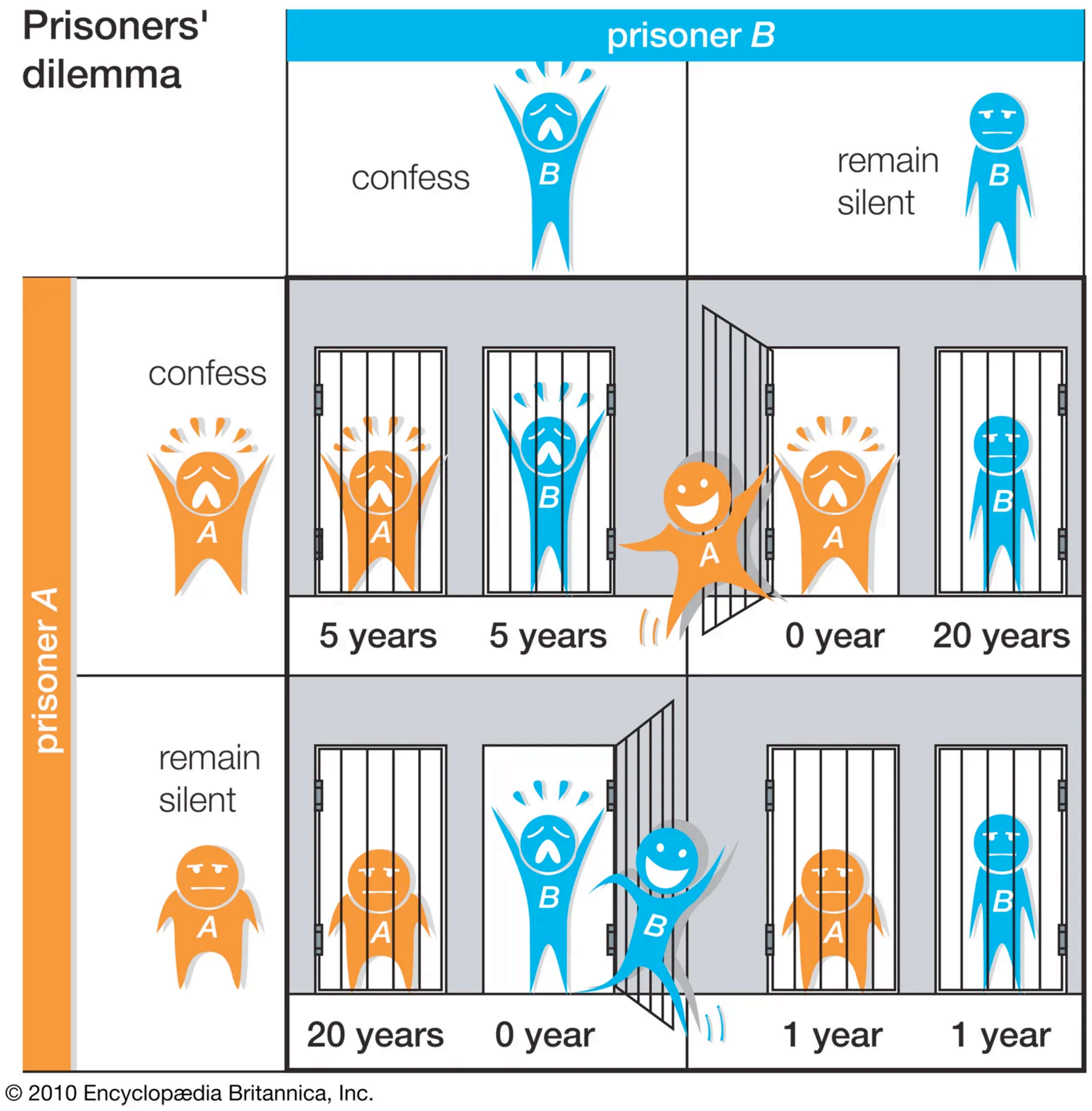
Nash equilibria do not need to be Pareto optimal.

- The problem might not be due to selfishness or lack of altruism.



Nash equilibria do <u>not</u> need to be Pareto optimal.

- The problem might not be due to selfishness or lack of altruism.
  - Players may not know enough.



Nash equilibria do <u>not</u> need to be Pareto optimal.

- The problem might not be due to selfishness or lack of altruism.
  - Players may not know enough.
  - They may not have the ability to communicate or coordinate.

Secret Santa: Gift Exchange

$$u_{\text{Alice}}(\mathbf{x}) \equiv u_{\text{Alice}}(x_{\text{Bob}})$$

Alice's utility depends fully on Bob's choice.

Secret Santa: Gift Exchange

$$u_{\text{Alice}}(\mathbf{x}) \equiv u_{\text{Alice}}(x_{\text{Bob}})$$

Alice's utility depends fully on Bob's choice.

Alice and Bob can't give the ideal gift (maximize the other's utility) if they don't know what the other wants!

Mechanism Design

There's a whole area of game theory devoted to changing the structure of the game so that collectively efficient decisions may be reached individually.

(We won't go into this in this talk.)

Learning in Games

Instead, we go into another area of game theory studying **how/whether players reach a Nash equilibrium** and if they do, **which ones do they end up at**.

II. Learning in Games

Nash Equilibrium

Definition. A joint strategy $\mathbf{x}^\star$ is a **Nash equilibrium** if no player can unilaterally improve their own utility:

$$u_n(\mathbf{x}^\star) = \max_{x_n \in \Omega_n} u_n(x_n; \mathbf{x}^\star_{-n}).$$

Why do we care about Nash equilibria? It is a minimal solution concept:

If a joint strategy $\mathbf{x}$ is not Nash, then it is not stable.

- At least one player could optimize further, causing the system to diverge from $\mathbf{x}$.
- Collective behavior with long-term viability "must" be Nash.

Nash Equilibrium

Definition. A joint strategy $\mathbf{x}^\star$ is a **Nash equilibrium** if no player can unilaterally improve their own utility:

$$u_n(\mathbf{x}^\star) = \max_{x_n \in \Omega_n} u_n(x_n; \mathbf{x}^\star_{-n}).$$

Why do we care about Nash equilibria? It is a minimal solution concept:

If a joint strategy $\mathbf{x}$ is not Nash, then it is not stable.

- At least one player could optimize further, causing the system to diverge from $\mathbf{x}$.
- Collective behavior with long-term viability “must” be Nash.

Nash Equilibrium

Definition. A joint strategy $\mathbf{x}^\star$ is a Nash equilibrium if no player can unilaterally improve their own utility:

$$u_n(\mathbf{x}^\star) = \max_{x_n \in \Omega_n} u_n(x_n; \mathbf{x}^\star_{-n}).$$

Why do we care about Nash equilibria? It is a minimal solution concept:

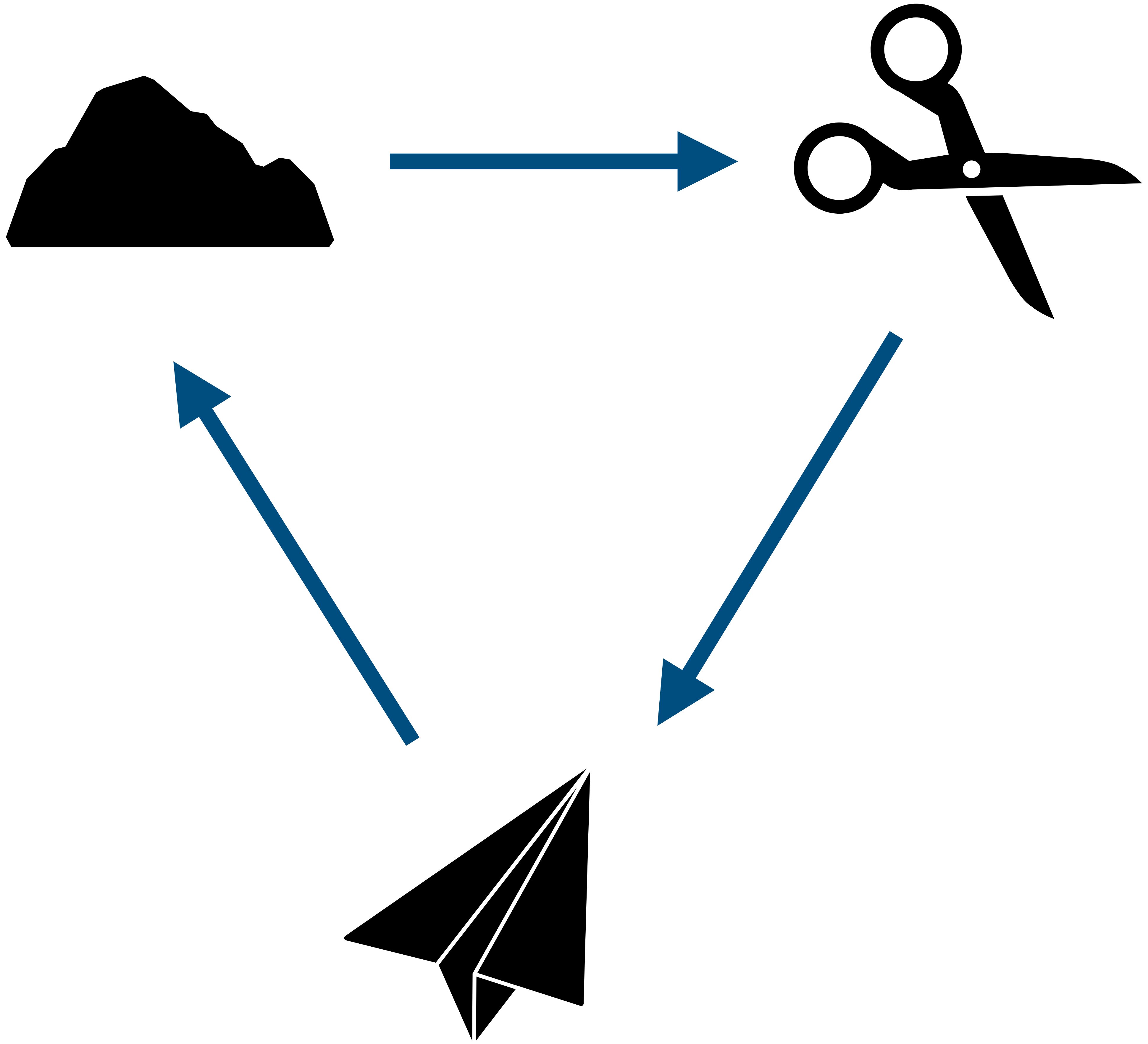
If a joint strategy $\mathbf{x}$ is not Nash, then it is not stable.

- At least one player could optimize further, causing the system to diverge from $\mathbf{x}$.
- Collective behavior with long-term viability “must” be Nash.

Issues with the Nash Equilibrium

1. **Existence**
2. **Uniqueness**
3. **Computability**
4. **Stability**

1. Existence



Rock-Paper-Scissors does not have a Nash equilibrium unless we allow **randomized strategies**.

Normal-Form Games

Each player has a finite set of **alternatives** $\mathscr{A}_n = \{a_n^{(1)}, \ldots, a_n^{(K_n)}\}$

Normal-Form Games

Each player has a finite set of alternatives $\mathscr{A}_n = \{a_n^{(1)}, \ldots, a_n^{(K_n)}\}$

- **For each** $\mathbf{a} = (a_1, \ldots, a_N) \in \mathscr{A}_1 \times \cdots \times \mathscr{A}_N$**, the utility is**

$$u_n(\mathbf{a}) = U_n^{a_1, \ldots, a_N}$$

Normal-Form Games

Each player has a finite set of alternatives $\mathscr{A}_n = \{a_n^{(1)}, \ldots, a_n^{(K_n)}\}$

- For each $\mathbf{a} = (a_1, \ldots, a_N) \in \mathscr{A}_1 \times \cdots \times \mathscr{A}_N$, the utility is

$$u_n(\mathbf{a}) = U_n^{a_1, \ldots, a_N}$$

- We allow for randomized strategies $x_n \in \Delta(\mathscr{A}_n)$

Normal-Form Games

Each player has a finite set of **alternatives** $\mathscr{A}_n = \{a_n^{(1)}, \ldots, a_n^{(K_n)}\}$

- For each $\mathbf{a} = (a_1, \ldots, a_N) \in \mathscr{A}_1 \times \cdots \times \mathscr{A}_N$, the utility is

$$u_n(\mathbf{a}) = U_n^{a_1, \ldots, a_N}$$

- We allow for **randomized strategies** $x_n \in \Delta(\mathscr{A}_n)$
  - The **mixed strategy** x_n describes a distribution over $\mathscr{A}_n$

Normal-Form Games

Each player has a finite set of alternatives $\mathscr{A}_n = \{a_n^{(1)}, \ldots, a_n^{(K_n)}\}$

- For each $\mathbf{a} = (a_1, \ldots, a_N) \in \mathscr{A}_1 \times \cdots \times \mathscr{A}_N$, the utility is

$$u_n(\mathbf{a}) = U_n^{a_1, \ldots, a_N}$$

- We allow for randomized strategies $x_n \in \Delta(\mathscr{A}_n)$
  - The mixed strategy x_n describes a distribution over $\mathscr{A}_n$
  - Let u_n be the expected utility when players randomize independently

Normal-Form Games

Each player has a finite set of alternatives $\mathscr{A}_n = \{a_n^{(1)}, \ldots, a_n^{(K_n)}\}$

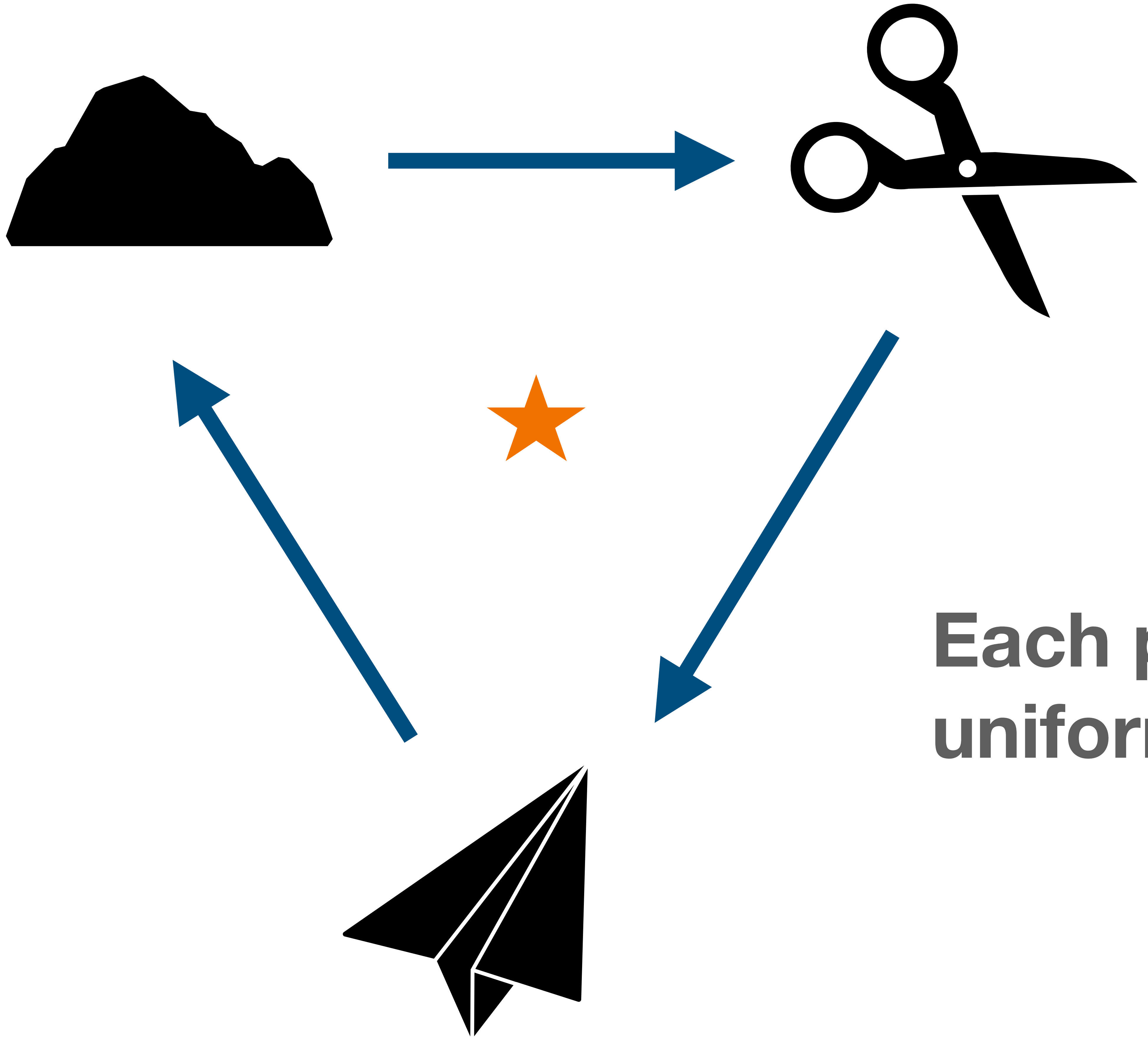
- For each $\mathbf{a} = (a_1, \ldots, a_N) \in \mathscr{A}_1 \times \cdots \times \mathscr{A}_N$, the utility is

$$u_n(\mathbf{a}) = U_n^{a_1, \ldots, a_N}$$

- We allow for randomized strategies $x_n \in \Delta(\mathscr{A}_n)$
  - The mixed strategy x_n describes a distribution over $\mathscr{A}_n$
  - Let u_n be the expected utility when players randomize independently

$$u_n(\mathbf{x}) = \mathbb{E}_{a_i \sim x_i}\left[U_n^{a_1, \ldots, a_N}\right]$$

Mixed Equilibrium



Each player picks R/P/S uniformly at random.

Minimax Theorem & Nash Existence

Existence of mixed equilibria in zero-sum and general normal-form games.

Minimax Theorem & Nash Existence

Existence of mixed equilibria in zero-sum and general normal-form games.

"As far as I can see, there could be no theory of games …
without that theorem.

I thought there was nothing worth publishing until the
Minimax Theorem was proved."

— John von Neumann (1928)

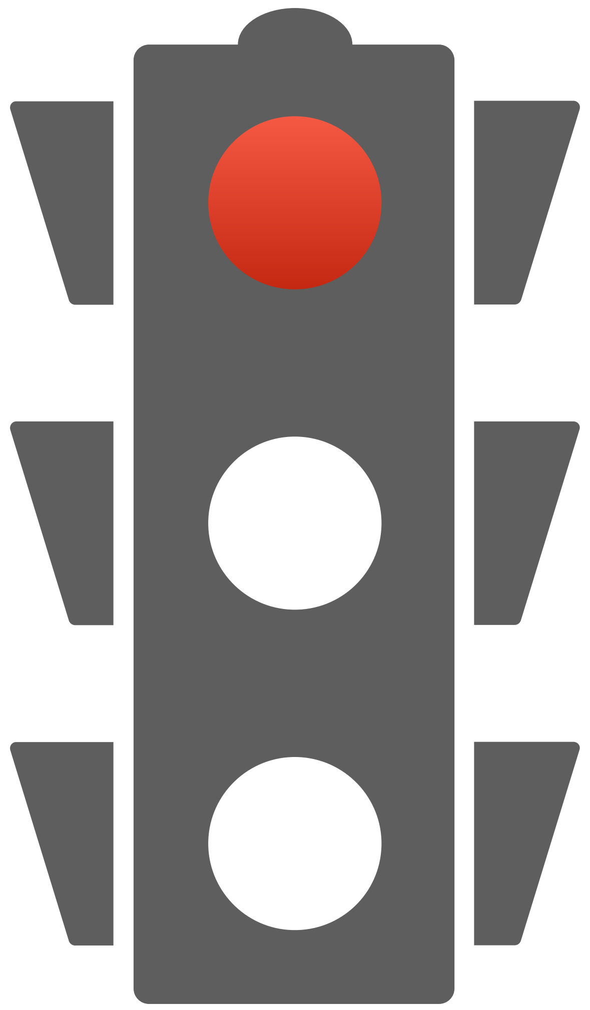
Minimax Theorem & Nash Existence

Existence of mixed equilibria in zero-sum and general normal-form games.

> “As far as I can see, there could be no theory of games …
> without that theorem.
>
> I thought there was nothing worth publishing until the
> Minimax Theorem was proved.”
>
> — John von Neumann (1928)

Implication for algorithms with worst-case guarantees: randomization may be necessary!

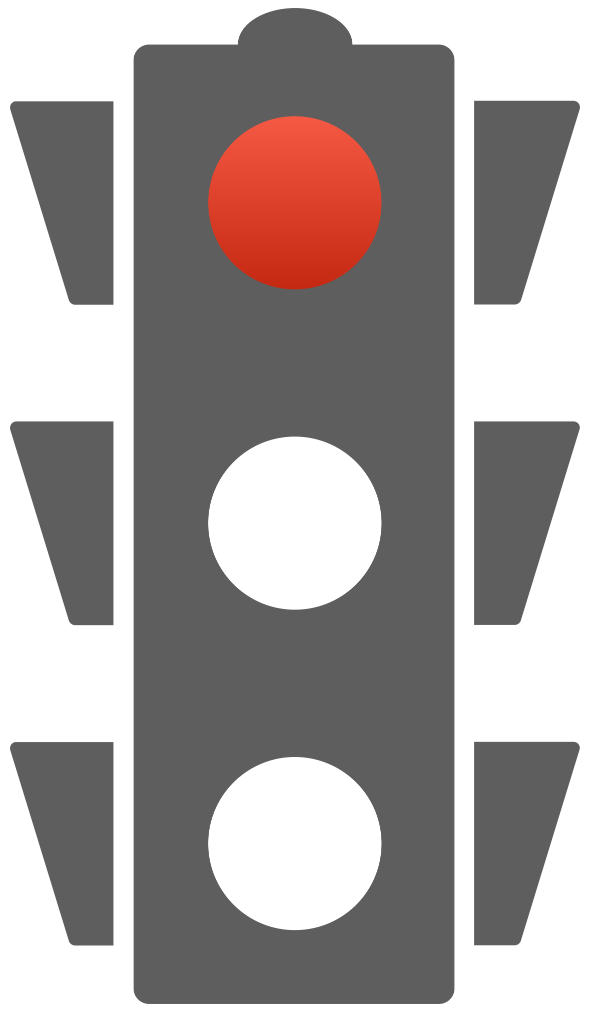
2. Uniqueness



Multiple Nash equilibria:

A. (Red, Green) = (Stop, Go)

2. Uniqueness

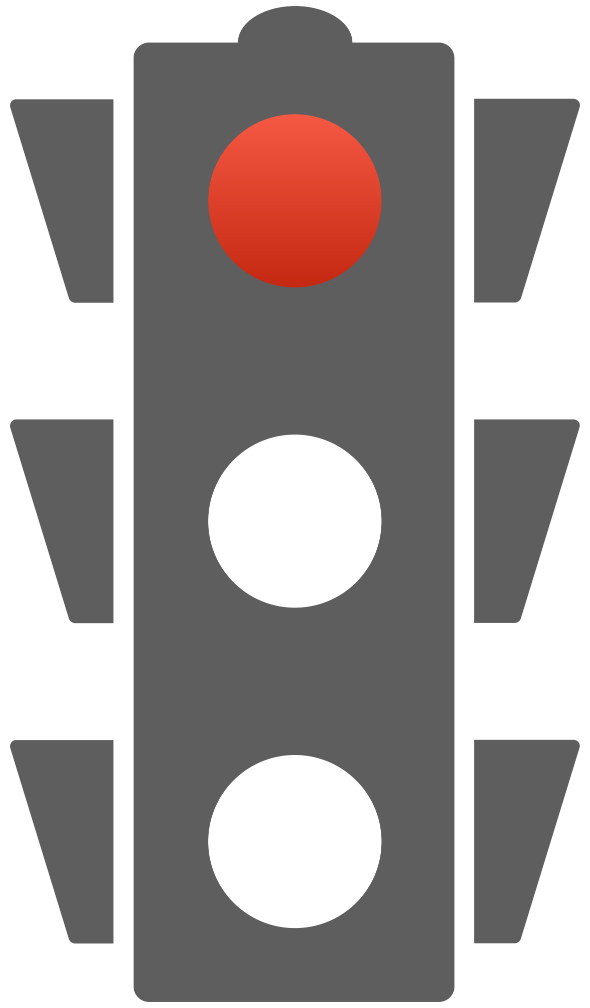


Multiple Nash equilibria:

A. (Red, Green) = (Stop, Go)

B. (Red, Green) = (Go, Stop)

2. Uniqueness



Multiple Nash equilibria:

A. (Red, Green) = (Stop, Go)

B. (Red, Green) = (Go, Stop)

C. Ignore signal and go with low probability.

The Equilibrium Selection Problem

If there are many equilibria, how do players know which one to play?

> “The Nash equilibrium only makes sense if each player knows which strategies the others are playing; if the equilibrium recommended by the theory is not unique, the players will not have this knowledge.”
>
> — Robert Aumann

The Equilibrium Selection Problem

Solution: develop a model of **how players arrive at the equilibrium**.

The Equilibrium Selection Problem

Solution: develop a model of how players arrive at the equilibrium.

- We will need to introduce a *dynamical* aspect of games: how do players use experience to <u>learn to play the game</u>?

The Equilibrium Selection Problem

Solution: develop a model of how players arrive at the equilibrium.

- We will need to introduce a *dynamical* aspect of games: how do players use experience to learn to play the game?
- In economics, this may lead to fraught issues of "rationality".

The Equilibrium Selection Problem

Solution: develop a model of how players arrive at the equilibrium.

- We will need to introduce a *dynamical* aspect of games: how do players use experience to learn to play the game?
- In economics, this may lead to fraught issues of “rationality”.
- In computer science, we often get to define how agents learn.

3. Computability

The problem of finding a Nash equilibrium is PPAD[1] complete.

[1]Polynomial Parity Arguments on Directed graphs

"If your laptop cannot find [the Nash equilibrium], neither can the market."

— Kamal Jain

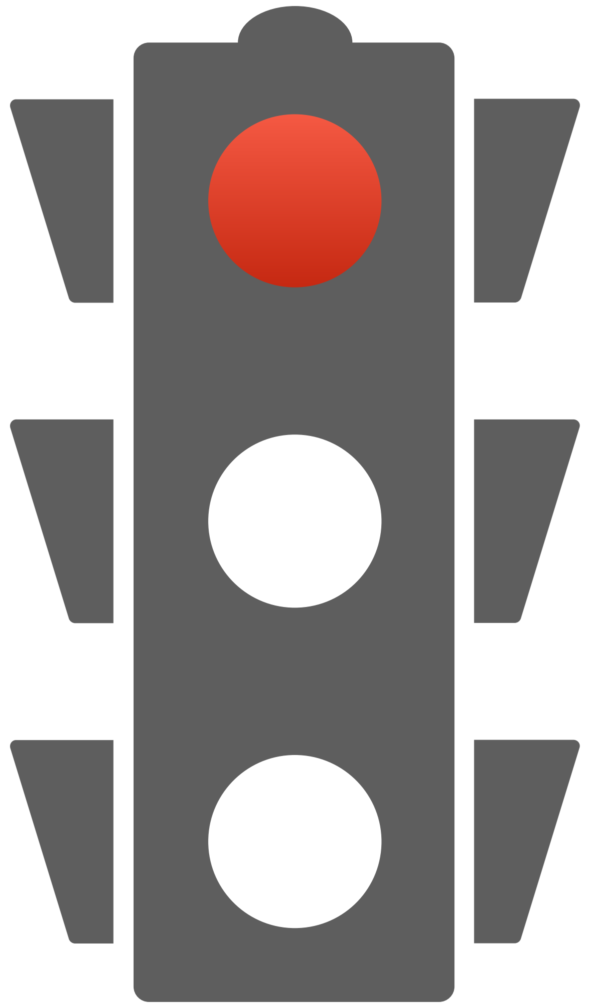
3. Computability (as players)

There exist games, for which all game dynamics fail to converge to Nash equilibria from all starting points.

(Milionis, Papadimitriou, Piliouras, Spendlove 2023)

(We won't go into computability in this talk.)

4. Stability



Multiple Nash equilibria:

A. (Red, Green) = (Stop, Go)

B. (Red, Green) = (Go, Stop)

C. Ignore signal and go with low probability.

This mixed equilibrium is **not stable**:

Once people start going slightly more often on green, the system will quickly converge on (Red, Green) = (Stop, Go).

Learning in Games

Goal: understand the dynamics of players learning to play a game

Learning in Games

Goal: understand the dynamics of players learning to play a game

- The equilibrium arises out of the dynamics (the players are not *trying* to compute it *per se*).

Learning in Games

Goal: understand the dynamics of players learning to play a game

- The equilibrium arises out of the dynamics (the players are not *trying* to compute it *per se*).
- Which equilibria can be robustly learned by players?

III. Simple Models of Learning

Repeated Games

Repeated Games

Players enter a game with limited knowledge:

- They know what set of actions they/others can take.
- They know their own utilities.

Repeated Games

Players enter a game with limited knowledge:

- They know what set of actions they/others can take.
- They know their own utilities.

They do not know:

- What other players want.
- How other players learn.

Repeated Games

Players enter a game with limited knowledge:

- They know what set of actions they/others can take.
- They know their own utilities.

They do not know:

- What other players want.
- How other players learn.

] The players are **uncoupled**.

Repeated Games

Players enter a game with limited knowledge:

- They know what set of actions they/others can take.
- They know their own utilities.

They do not know:

- What other players want.
- How other players learn.

Players will repeatedly play the game, observe what other do, and adjust their own behaviors for future rounds.

Repeated Games

Players enter a game with limited knowledge:

- They know what set of actions they/others can take.
- They know their own utilities.

They do not know:

- What other players want.
- How other players learn.

Players will repeatedly play the game, observe what other do, and adjust their own behaviors for future rounds.

A **learning rule** describes how a player make use of past experience.

Simple Learning Rules

- **Best Response:** optimize, assuming that players in the next round will continue to play the current strategy.

Simple Learning Rules

- **Best Response:** optimize, assuming that players in the next round will continue to play the current strategy.
- **Follow the Leader:** optimize, assuming that past experience forms a representative sample of strategies that will be played.

Simple Learning Rules

- **Best Response:** optimize, assuming that players in the next round will continue to play the current strategy.
- **Follow the Leader:** optimize, assuming that past experience forms a representative sample of strategies that will be played.
- **No-Regret Learning:** apply standard online learning algorithms.

Learning rules define a dynamical system

The **dynamics** of a (Markov) learning rule is governed by a transition map:

$$T : \mathbf{\Omega} \to \mathbf{\Omega},$$

where $\mathbf{x}(t+1) = T\big(\mathbf{x}(t)\big)$.

Learning rules define a dynamical system

The **dynamics** of a (Markov) learning rule is governed by a transition map:

$$T : \mathbf{\Omega} \to \mathbf{\Omega},$$

where $\mathbf{x}(t+1) = T\big(\mathbf{x}(t)\big)$.

- A joint strategy $\mathbf{x}^\dagger$ is an **equilibrium** or a **fixed point** if:

$$\mathbf{x}^\dagger = T(\mathbf{x}^\dagger).$$

Best-Response Dynamics

The **best-response map** defines the learning rule:

$$T_n(\mathbf{x}) = \arg\max_{x'_n \in \Omega_n} \; u_n\left(x'_n; \mathbf{x}_{-n}\right).$$

Best-Response Dynamics

The **best-response map** defines the learning rule:

$$T_n(\mathbf{x}) = \arg\max_{x_n' \in \Omega_n} u_n\left(x_n'; \mathbf{x}_{-n}\right).$$

- A Nash equilibrium is a fixed point of the best-response map.

Gradient Ascent-Ascent: a learning rule

Assume that utilities are smooth: $u_n : \Omega_n \to \mathbb{R}$.

For $t = 1,2,\ldots$:

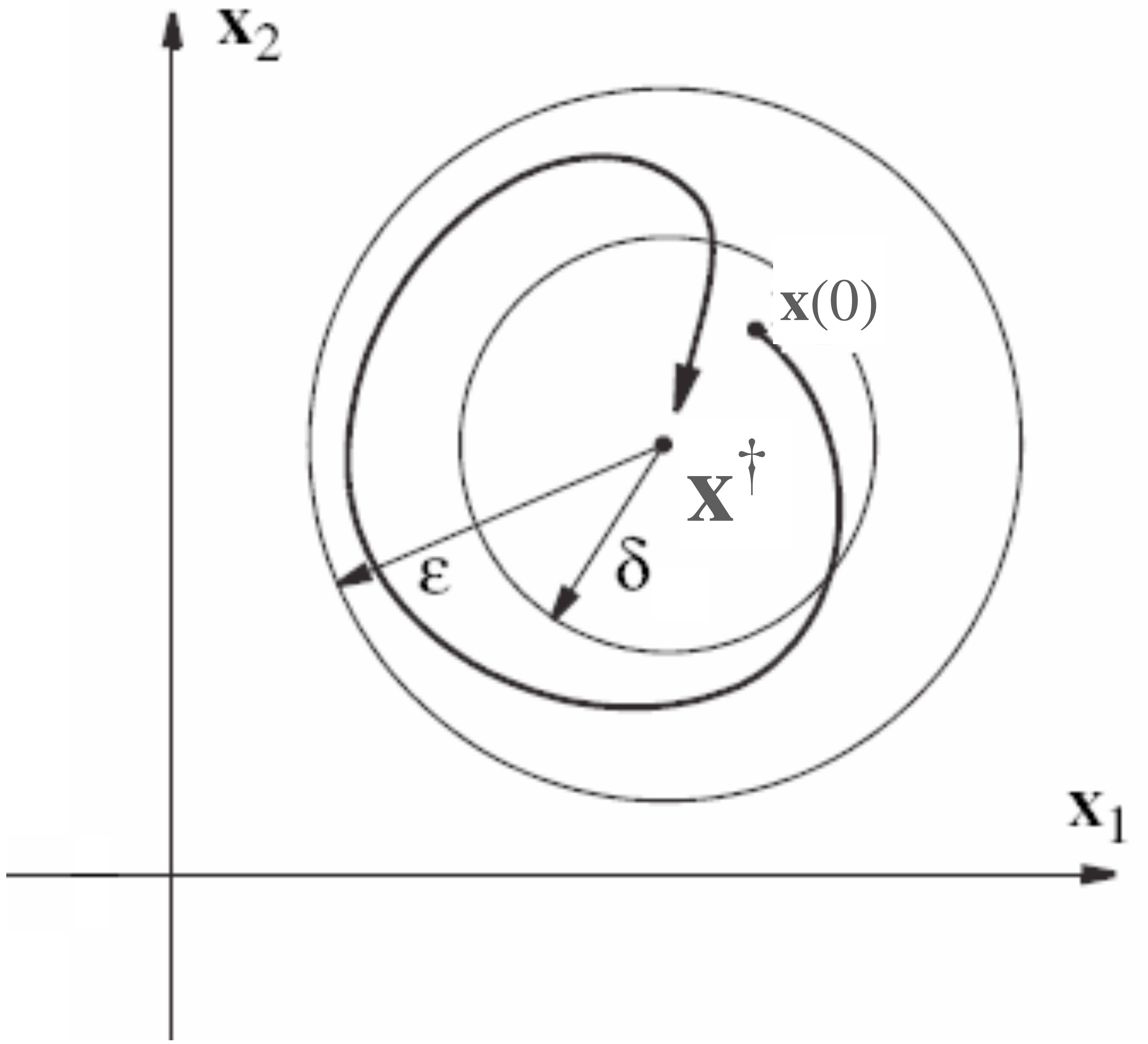
- **Players update their strategies by gradient ascent:**

$$x_n(t+1) = x_n(t) + \eta \cdot \nabla_n u_n\big(\mathbf{x}(t)\big)$$

Classic Notions of Dynamical Stability

1. Asymptotic stability
2. Non-asymptotic/Lyapunov stability
3. Instability

1. Asymptotic Stability



An equilibrium $\mathbf{x}^\dagger$ is **asymptotically stable** if there is some $\delta > 0$ such that:

$$\|\mathbf{x}(0) - \mathbf{x}^\dagger\| < \delta \quad \Longrightarrow \quad \lim_{t\to\infty} \mathbf{x}(t) = \mathbf{x}^\dagger$$

A strong notion of stability: convergence toward equilibrium is stable against small perturbations.

Asymptotic Stability for Strict NE

Strict Nash equilibria admit asymptotically stable dynamics.

Asymptotic Stability for Strict NE

Strict Nash equilibria admit asymptotically stable dynamics.

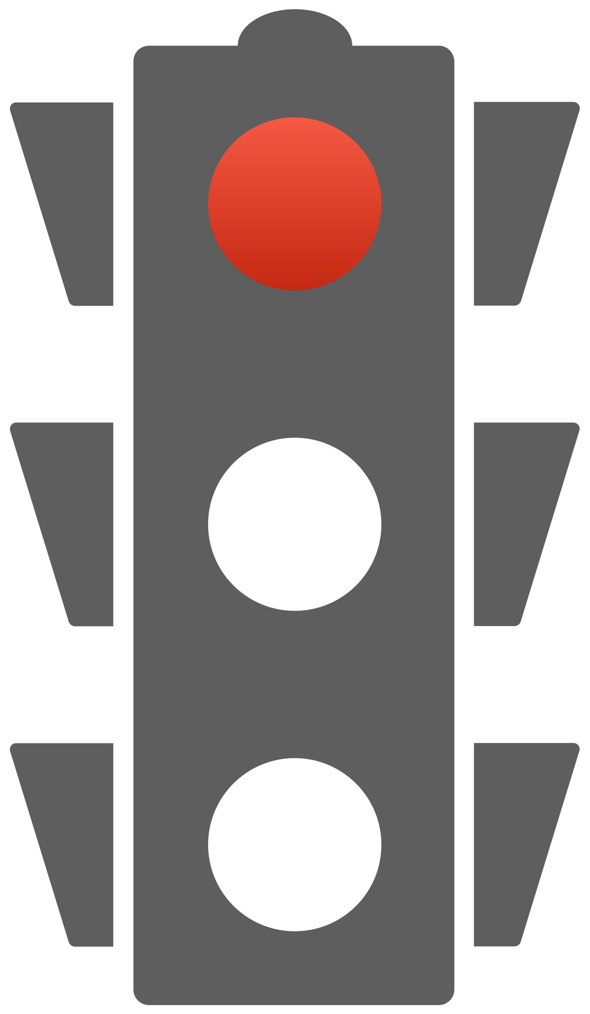
Given that others play $\mathbf{x}^\star_{-n}$, there is a clear, deterministic best response $x_n^\star$.

Asymptotic Stability for Strict NE

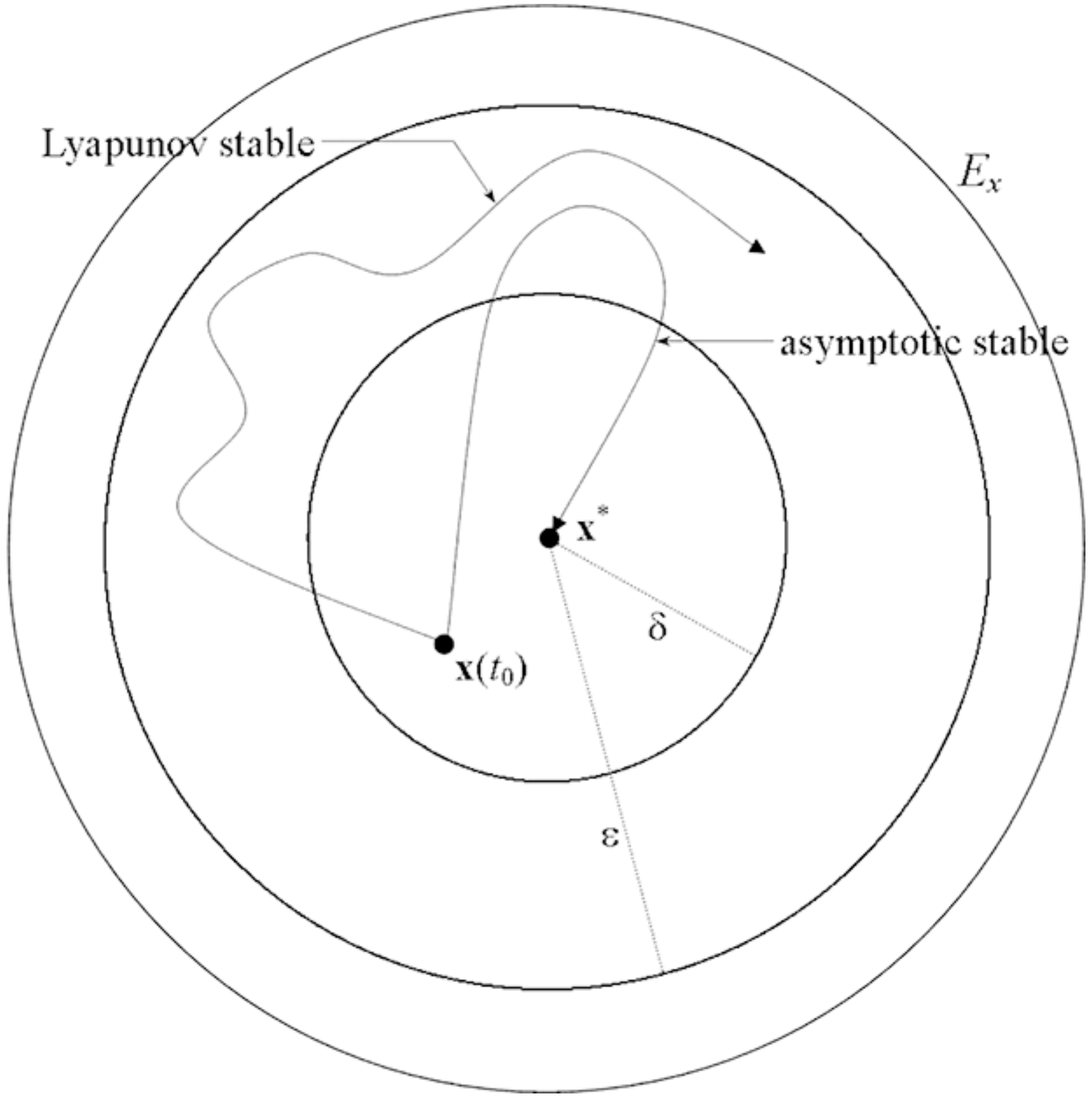
Strict Nash equilibria admit **asymptotically stable** dynamics.

Given that others play $\mathbf{x}^\star_{-n}$, there is a clear, deterministic best response $x^\star_n$.

The traffic light game is an example of this.



2. Lyapunov (Non-Asymptotic) Stability

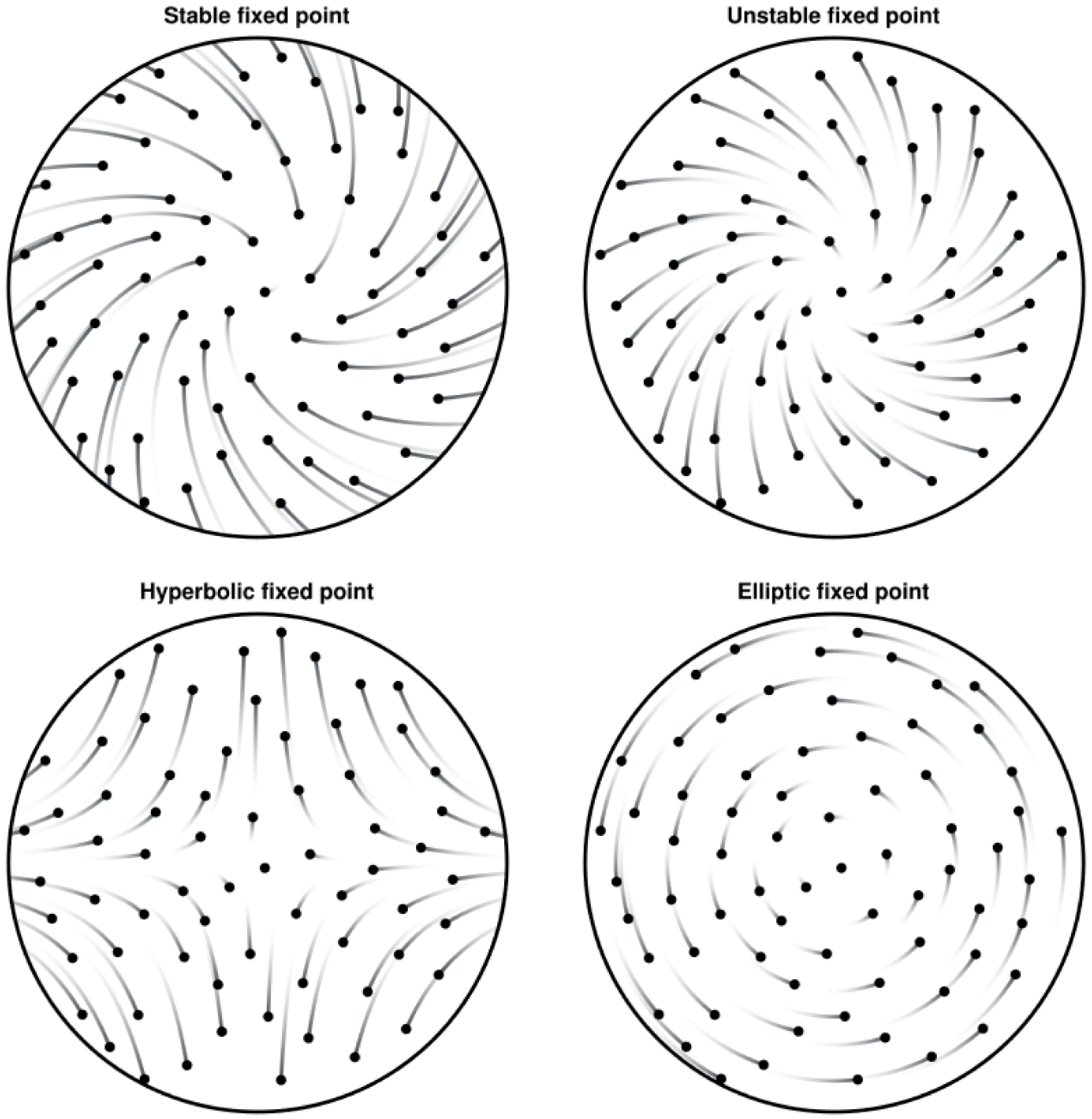


An equilibrium $\mathbf{x}^\dagger$ is Lyapunov stable if for all $\varepsilon > 0$, there is some $\delta > 0$ such that:

$$\|\mathbf{x}(0) - \mathbf{x}^\dagger\| < \delta \quad \Longrightarrow \quad \sup_{t>0} \|\mathbf{x}(t) - \mathbf{x}^\dagger\| < \varepsilon$$

A weaker notion of stability: dynamics near equilibrium remains close to it.

3. Unstable



An equilibrium $\mathbf{x}^{\dagger}$ is unstable if it is not Lyapunov stable.

Asymptotically-Stable Uncoupled Dynamics

Theorem (Hart & Mas-Colell 2003). Uncoupled dynamics cannot achieve asymptotically-stable convergence to all Nash equilibria.

Read: due to informational constraints, not all Nash equilibria are "strongly learnable".

Instability for non-strict/mixed NE

Many learning dynamics exhibit **instability** around **mixed** NE.

↑

It is troubling for the theoretical viability of mixed NE, if it is the case that players cannot converge to it.

Convergence in Zero-Sum Games

However, there are uncoupled dynamics that do robustly find certain equilibria (e.g. those in zero-sum games):

- Optimistic gradient descent-ascent
- Extragradient method
- Smoothed best-response

This Work

Which mixed equilibria admit convergence?

Learnable Mixed Nash Equilibria are Collectively Rational

Geelon So and Yi-An Ma, Preprint 2025

IV. Uniform Stability and Collective Rationality

Linear Stability Analysis

Consider the following continuous-time, smooth dynamical system:

$$\dot{\mathbf{x}}(t) = T\Big(\mathbf{x}(t)\Big).$$

Linear Stability Analysis

Consider the following continuous-time, smooth dynamical system:

$$\dot{\mathbf{x}}(t) = T\Big(\mathbf{x}(t)\Big).$$

- Example: $\dot{x}_n(t) = \nabla_n u_n\Big(\mathbf{x}(t)\Big)$

Linear Stability Analysis

Consider the following continuous-time, smooth dynamical system:

$$\dot{\mathbf{x}}(t) = T\big(\mathbf{x}(t)\big).$$

- Example: $\dot{x}_n(t) = \nabla_n u_n\big(\mathbf{x}(t)\big)$
- We can study the **stability** of the dynamics around a fixed point $\mathbf{x}^\dagger$ by looking at the **linearized dynamics** around $\mathbf{x}^\dagger$.

Linearized Dynamics

For simplicity, let $\mathbf{x} \in \mathbb{R}^N$. The **Jacobian** $\mathbf{J}(\mathbf{x}) \in \mathbb{R}^{N \times N}$ is given by:

$$\mathbf{J} = \nabla T = \begin{bmatrix} \nabla_1 T_1 & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \ddots & & \\ \vdots & & & \\ \nabla_N T_1 & & & \end{bmatrix}$$

Linearized Dynamics

For simplicity, let $\mathbf{x} \in \mathbb{R}^N$. The **Jacobian** $\mathbf{J}(\mathbf{x}) \in \mathbb{R}^{N \times N}$ is given by:

$$\mathbf{J} = \nabla T = \begin{bmatrix} \nabla_1 T_1 & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \ddots & & \\ \vdots & & & \\ \nabla_N T_1 & & & \end{bmatrix}$$

Around the fixed point, the linearized dynamics are defined by:

$$\dot{\mathbf{z}}(t) = \mathbf{J}(\mathbf{x}^{\dagger})\,\mathbf{z}(t)$$

where $\mathbf{z} \equiv \mathbf{x} - \mathbf{x}^{\dagger}$.

Stability from Spectral Analysis

Stability of **linear** dynamical systems well-understood:

- If all eigenvalues of $\mathbf{J}$ have **negative real parts** $\implies$ asymptotic stability
- If there is an eigenvalue with **positive real part** $\implies$ instability
- If all eigenvalues are **purely imaginary** $\implies$ neutral stability
  - This case is much harder to characterize for non-linear systems

Barrier to Asymptotic Stability

The **Jacobian** of many game dynamics look this:

$$\mathbf{J}(\mathbf{x}^\dagger) = \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

Barrier to Asymptotic Stability

The **Jacobian** of many game dynamics look this:

$$\mathbf{J}(\mathbf{x}^\dagger) = \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

- The dynamics are **traceless:** $\mathrm{tr}\Big(\mathbf{J}(\mathbf{x}^\dagger)\Big) = 0.$

Barrier to Asymptotic Stability

The **Jacobian** of many game dynamics look this:

$$\mathbf{J}(\mathbf{x}^\dagger) = \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

- The dynamics are **traceless:** $\mathrm{tr}\left(\mathbf{J}(\mathbf{x}^\dagger)\right) = 0.$
  - But, the sum of eigenvalues of $\mathbf{J}$ is the trace:

$$\mathrm{tr}\left(\mathbf{J}\right) = \lambda_1 + \cdots + \lambda_N \,.$$

Barrier to Asymptotic Stability

The **Jacobian** of many game dynamics look this:

$$\mathbf{J}(\mathbf{x}^\dagger) = \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

- The dynamics are **traceless:** $\mathrm{tr}\big(\mathbf{J}(\mathbf{x}^\dagger)\big) = 0.$
  - But, the sum of eigenvalues of $\mathbf{J}$ is the trace:

$$\mathrm{tr}\big(\mathbf{J}\big) = \lambda_1 + \cdots + \lambda_N \,.$$

- Asymptotic stability requires that $\Re(\lambda_n) < 0$ for all $n \in [N]$. This is impossible for traceless dynamics.

Not unstable $\implies$ purely imaginary eigenvalues

Jacobians of uncoupled game dynamics

The **Jacobian** of many **uncoupled** game dynamics look this:

$$\mathbf{J}(\mathbf{x}^\dagger; \mathbf{H}) = \begin{bmatrix} H_1 & & & \\ & H_2 & & \\ & & \ddots & \\ & & & H_N \end{bmatrix} \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

Jacobians of uncoupled game dynamics

The **Jacobian** of many **uncoupled** game dynamics look this:

$$\mathbf{J}(\mathbf{x}^{\dagger};\mathbf{H}) = \begin{bmatrix} H_1 & & & \\ & H_2 & & \\ & & \ddots & \\ & & & H_N \end{bmatrix} \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

where H_n are positive-definite matrices (think: preconditioners, player-dependent geometry).

Jacobians of uncoupled game dynamics

The **Jacobian** of many **uncoupled** game dynamics look this:

$$\mathbf{J}(\mathbf{x}^{\dagger}; \mathbf{H}) = \begin{bmatrix} H_1 & & & \\ & H_2 & & \\ & & \ddots & \\ & & & H_N \end{bmatrix} \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

where H_n are positive-definite matrices (think: preconditioners, player-dependent geometry).

- Example: $\dot{x}_n(t) = H_n \nabla_n u_n\big(\mathbf{x}(t)\big)$

Jacobians of uncoupled game dynamics

The **Jacobian** of many **uncoupled** game dynamics look this:

$$\mathbf{J}(\mathbf{x}^\dagger;\mathbf{H}) = \begin{bmatrix} H_1 & & & \\ & H_2 & & \\ & & \ddots & \\ & & & H_N \end{bmatrix} \begin{bmatrix} \mathbf{0} & \nabla_2 T_1 & \cdots & \nabla_N T_1 \\ \nabla_1 T_2 & \mathbf{0} & & \\ \vdots & & \ddots & \\ \nabla_N T_1 & & & \end{bmatrix}$$

where H_n are positive-definite matrices (think: preconditioners, player-dependent geometry).

- Example: $\dot{x}_n(t) = H_n \nabla_n u_n\big(\mathbf{x}(t)\big)$
- Stability for uncoupled dynamics **requires** all eigenvalues of $\mathbf{J}(\mathbf{x}^\dagger;\mathbf{H})$ be be purely imaginary.

Jacobians of uncoupled game dynamics

Definition. An equilibrium $\mathbf{x}^\dagger$ is **uniformly stable** if the game Jacobian $\mathbf{J}(\mathbf{x}^\dagger; \mathbf{H})$ has purely imaginary eigenvalues for all $\mathbf{H} = (H_1, \dots, H_n)$ where $H_n \succ 0$.

- It is **locally uniformly stable** if this holds for all $\mathbf{x}$ in an open set around $\mathbf{x}^\dagger$.

Main Result: Economic Meaning of Stability

Local Uniform Stability $\Longrightarrow$ Strategic Pareto Optimality

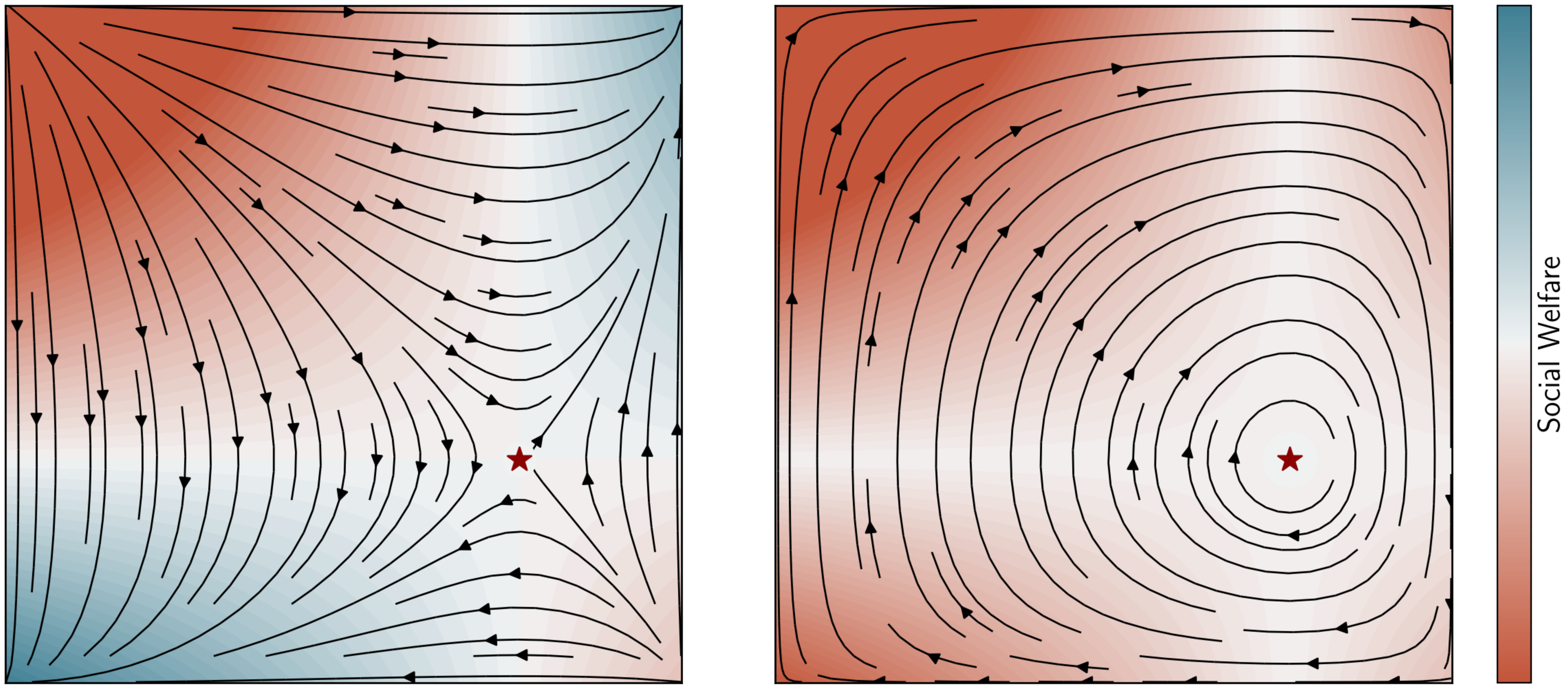
Strategic Pareto Optimality $\Longrightarrow$ Uniform Stability

Main Result: Uniform Stability and Convergence

Local Uniform Stability $\implies$ Dynamics can be stabilized

Not Uniformly Stable $\implies$ Dynamics cannot be stabilized

Comparison of Dynamics Around Non-Strict NE



Unstable Nash equilibria are not strategically Pareto optimal.

Uniformly stable Nash equilibria are strategically Pareto optimal.

Takeaways

- **Modern ML often implements multi-agent solutions.**
- **Decentralization introduces structural constraints.**
- **What are the ramifications and when are guardrails needed?**