

Learning to Control Unknown Strongly Monotone Games

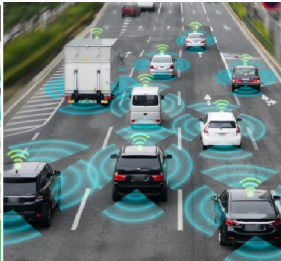
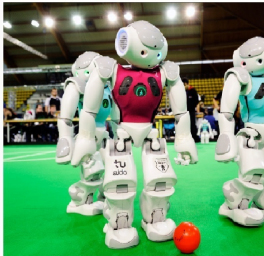
Siddharth Chandak

Joint work with Prof. Ilai Bistritz (Tel Aviv University) and Prof. Nicholas Bambos (Stanford University)

- Overview
 - Game Control
 - Strongly Monotone Games and Nash Equilibrium
- Equilibrium Steering
 - Steering via Linear Utility Parameters
 - Two-time-scale Stochastic Approximation

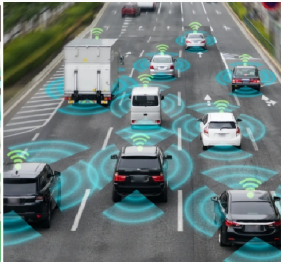
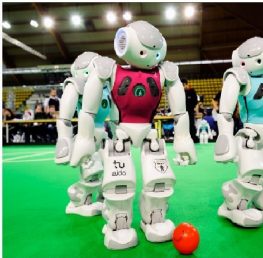
Overview

Multi-Agent Systems



Multi-Agent Games

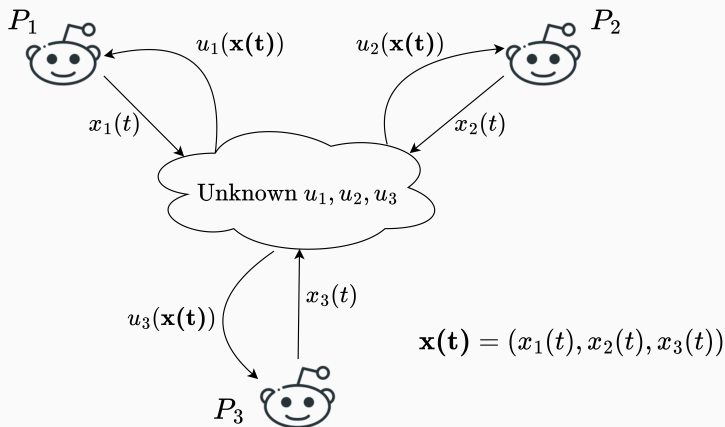
- Game with N agents or players
- Each player n takes action x_n
- Utility (Reward): $u_n(x_1, \dots, x_N)$



Local Objective

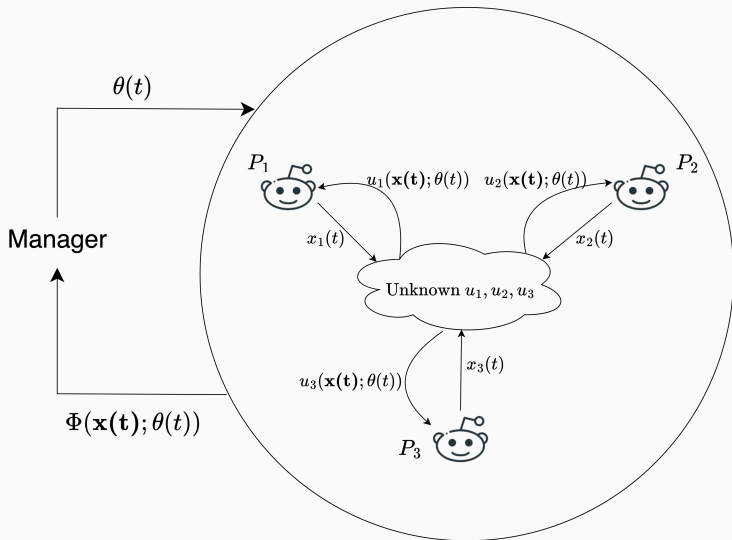
- **Local Objective:** Each player n wants to maximize their reward $u_n(\mathbf{x}_1, \dots, \mathbf{x}_N)$ under the constraint of limited feedback

Bandit Feedback



- Game Manager or System Controller
 - Control some parameter θ of the game
 - For example, can control the action set of players, or the utilities of players
- Have their own objective - the **“Global Objective”**
 - Each player is optimizing for the local objective of $u_n(\mathbf{x}; \theta)$
 - The manager is optimizing for the global objective of $\Phi(\mathbf{x}; \theta)$
 - Bandit feedback

Game Control



Evolution of Players' Actions

- How do players update their actions?
- Converge to Nash equilibrium?
- We focus on a class of games called **Strongly Monotone Games**

Strongly Monotone Games and Nash Equilibrium

Strongly Monotone Games

- Class of continuous action games
- Unique pure Nash Equilibrium (NE)
- Each player performing gradient ascent on their utilities leads to convergence to NE
 - Stronger than just convergence: geometric convergence
 - *Intuitively*: multi-agent extension of strongly concave functions

Definition

- Suppose player n chooses actions in $\mathcal{X}_n \subseteq \mathbb{R}^d$ where $\mathcal{X}_n$ is convex and compact
- Define the concatenated gradient operator $G(\cdot) : \mathbb{R}^{Nd} \mapsto \mathbb{R}^{Nd}$ as

$$G(\mathbf{x}) = (\nabla_{x_1} u_1(x_1, \mathbf{x}_{-1}), \dots, \nabla_{x_N} u_N(x_N, \mathbf{x}_{-N})),$$

where $\mathbf{x} = (x_1, \dots, x_N)$

Definition 1 (Strongly Monotone Games)

There exists $\mu > 0$ such that for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$,

$$\langle \mathbf{y} - \mathbf{x}, G(\mathbf{y}) - G(\mathbf{x}) \rangle \leq -\mu \|\mathbf{y} - \mathbf{x}\|^2$$

- Suppose each player updates their actions as follows (for stepsize η_t):

$$x_{n,t+1} = x_{n,t} + \eta_t \nabla_{x_n} u_n(x_{n,t}, \mathbf{x}_{-n,t})$$

- Converges to unique pure NE $\mathbf{x}^*$

Definition 2

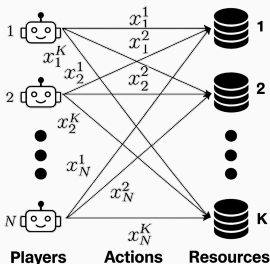
An action profile $\mathbf{x}^*$ is a pure Nash equilibrium (NE) if $u_n(x_n^*, \mathbf{x}_{-n}^*) \geq u_n(x_n, \mathbf{x}_{-n}^*)$, for all $x_n \in \mathcal{X}_n$ and all $n \in \mathcal{N}$.

Is this NE what we want?

- A NE is not always *desirable*
- Issues:
 - Inequality
 - Inefficiency - Braess' Paradox
 - Operational Issues - Resource Allocation Games

Resource Allocation Games

- K resources
- Each player's action is K –dimensional, where the i^{th} dimension represents the amount of i^{th} resource they use
- Example: electricity grids and wireless channels
- At NE - some resources are heavily used, creating pressure on system



	Hour 1	...	Hour 12	...	Hour 24
Player 1	250 W	...	1000 W	...	100 W
Player 2	150 W	...	800 W	...	50 W
⋮	⋮		⋮		⋮
Player N	400 W	...	1500 W	...	250 W

A Controlled Strongly Monotone Game

- Recall that utilities are given by $u_n(\mathbf{x}; \theta)$
- Players update their actions using gradient ascent

$$x_{n,t+1} = x_{n,t} + \eta_t \nabla_{x_n} u_n(\mathbf{x}_t; \theta_t)$$

- For fixed θ , players converge to some $\mathbf{x}^*(\theta)$

- **Problem Statement:** How to choose the control θ_t such that the players converge to a desirable NE under noisy bandit feedback?

Linear Utility Parameters

Linear Coefficients

- Each player takes action $x_n = (x_n^{(1)}, \dots, x_n^{(d)})$ in a compact and convex set $\mathcal{X}_n \subseteq \mathbb{R}^d$
- Utility for each player is given by:

$$u_n(\mathbf{x}, \beta_n^{(1)}, \dots, \beta_n^{(d)}) = r_n(\mathbf{x}) - \sum_{i=1}^d \beta_n^{(i)} x_n^{(i)}$$

- $r_n(x)$ - reward from 'original' uncontrolled game without any control
- $\sum_{i=1}^d \beta_n^{(i)} x_n^{(i)}$ - linear shift in utility

Control Parameter and Manager's Objective

- $\sum_{i=1}^d \beta_n^{(i)} x_n^{(i)}$ - linear shift in utility
- The controllable game parameter θ is the Nd -dimensional vector β
- Steer the players' NE towards a point that satisfies K linear constraints:

$$A\mathbf{x} = \ell^*$$

- Manager only observes the constraint violation $A\mathbf{x}_t - \ell^*$

Application: Resource Allocation

- Recall that $x_n^{(i)}$ denotes how much player n uses resource i
- Suppose the target constraints are of the form

$$\sum_{n=1}^N x_n^{(i)} = \ell_i^*$$

for each resource $i \in \{1, \dots, K\}$

- Then the manager can set β_i for each resource i (constant across all players)
 - Additional price or subsidy on using a resource

Assumptions and Problem Formulation

- The uncontrolled game with utilities $r_n(\mathbf{x})$ is strongly monotone
 - Let $F(\mathbf{x}) := (\nabla_{x_1} r_1(x_1, \mathbf{x}_{-1}), \dots, \nabla_{x_N} r_N(x_N, \mathbf{x}_{-N}))$

$$\langle \mathbf{y} - \mathbf{x}, F(\mathbf{y}) - F(\mathbf{x}) \rangle \leq -\mu \|\mathbf{y} - \mathbf{x}\|^2$$

- Gradient operator for controlled game is $G(\mathbf{x}) = F(\mathbf{x}) - \beta$
 - Implies that the controlled game is also strongly monotone
- Mapping $F(\cdot)$ is Lipschitz continuous
- At each timestep, player n observes noisy version of gradient of reward: $\nabla_{x_n} r_n(\mathbf{x}_t) + M_{n,t+1}$
 - $M_{n,t+1}$ is martingale difference noise with bounded second moment
- Slater's condition holds

Online Game Control Algorithm

Algorithm (Online Game Control)

Initialization: Let $x_0 \in \mathcal{X}$ and $\alpha_0 \in \mathbb{R}^K$.

For each turn $t \geq 0$ do

1. The manager broadcasts α_t to the players
2. The manager observes the vector $A\mathbf{x}_t - \ell^*$ and updates the controlled input using

$$\alpha_{t+1} = \alpha_t + \epsilon_t (A\mathbf{x}_t - \ell^*).$$

3. Each player n computes $\beta_{n,t} = A_n^T \alpha_t$ and updates its action using gradient ascent:

$$x_{n,t+1} = \Pi_{\mathcal{X}_n} (x_{n,t} + \eta_t (\nabla_{x_n} r_n(\mathbf{x}_t) + M_{n,t+1} - \beta_{n,t}))$$

where $\Pi_{\mathcal{X}_n}$ is the Euclidean projection into $\mathcal{X}_n$.

End

Understanding the Algorithm

- Vectorized Form:

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}} \left(\mathbf{x}_t + \eta_t \left(F(\mathbf{x}_t) - A^T \alpha_t + M_{t+1} \right) \right)$$

$$\alpha_{t+1} = \alpha_t + \epsilon_t (A \mathbf{x}_t - \ell^*)$$

- Instead of directly transmitting $\beta_t \in \mathbb{R}^{N_d}$, manager updates and transmits $\alpha_t \in \mathbb{R}^K$, such that $\beta_t = A^T \alpha_t$
- Iterative approach to solving the constrained optimization problem using Lagrange multipliers

Two-time-scale Stochastic Approximation (SA)

- Our algorithm is a two-time-scale stochastic approximation algorithm

$$\text{Faster: } \mathbf{x}_{t+1} = \Pi_{\mathcal{X}} \left(\mathbf{x}_t + \eta_t \left(F(\mathbf{x}_t) - A^T \alpha_t + M_{t+1} \right) \right)$$

$$\text{Slower: } \alpha_{t+1} = \alpha_t + \epsilon_t (A \mathbf{x}_t - \ell^*)$$

- Timescales dictated by stepsizes η_t and ϵ_t
 - η_t is larger, or decays at a slower rate, e.g., $1/n^{0.6}$
 - ϵ_t is smaller, or decays at a faster rate, e.g., $1/n^{0.75}$
- Intuition:
 - Faster time-scale: α_t considered quasi-static
 - Slower time-scale: $\mathbf{x}_t$ tracks $\mathbf{x}^*(\alpha_t)$, the NE corresponding to α_t

- Condition on stepsizes:

$$\eta_t = \frac{1}{(t + T_1)^\eta} \text{ and } \epsilon_t = \frac{1}{(t + T_2)^\epsilon},$$

where $0.5 < \eta < \epsilon < 1$. Importantly,

$$\frac{\epsilon_t^2}{\eta_t^3} \leq 1$$

Theorem

Define $\mathcal{N}_{opt} = \{\alpha \mid A\mathbf{x}^*(\alpha) = \ell^*\}$. Then

- α_t converges to the set $\mathcal{N}_{opt}$, $\mathbf{x}_t$ converges to $\mathbf{x}^*(\alpha_t)$, and

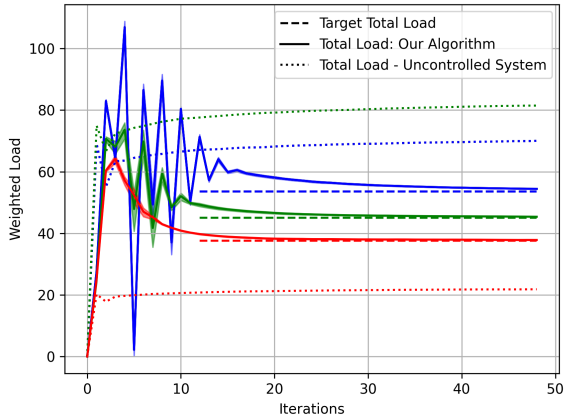
$$\lim_{t \rightarrow \infty} A\mathbf{x}_t = \ell^*,$$

with probability 1.

- $\mathbb{E}[\|A\mathbf{x}_t - \ell^*\|^2] = \mathcal{O}\left(\eta_t + \frac{1}{t\epsilon_t}\right).$

The best rate based on above result is $\mathcal{O}(t^{-0.25+\delta})$, where δ is arbitrarily small. This is achieved at $\eta = 0.5 + \delta/3$ and $\epsilon = 0.75 + \delta$.

Numerical Experiments: Power Distribution Grid



Utilities and loads based on H. L. Willis, *Power Distribution Planning Reference Book*, 2004.

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}} \left(\mathbf{x}_t + \eta_t (F(\mathbf{x}_t) - A^T \alpha_t + M_{t+1}) \right)$$

$$\alpha_{t+1} = \alpha_t + \epsilon_t (A\mathbf{x}_t - \ell^*)$$

- Can be expressed as fixed-point iterations:

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}} \left(\mathbf{x}_t + \eta_t (f(\mathbf{x}_t, \alpha_t) - \mathbf{x}_t + M_{t+1}) \right)$$

$$\alpha_{t+1} = \alpha_t + \epsilon_t (g(\alpha_t) - \alpha_t + \omega_t)$$

- Here

- $f(\mathbf{x}, \alpha) = \mathbf{x} + F(\mathbf{x}) - A^T \alpha$
- $g(\alpha) = \alpha + (A\mathbf{x}^*(\alpha) - \ell^*)$
- $\omega_t = A\mathbf{x}_t - A\mathbf{x}^*(\alpha_t)$ is the equilibrium noise

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}(\mathbf{x}_t + \eta_t(f(\mathbf{x}_t, \boldsymbol{\alpha}_t) - \mathbf{x}_t + M_{t+1}))$$

$$\boldsymbol{\alpha}_{t+1} = \boldsymbol{\alpha}_t + \epsilon_t(g(\boldsymbol{\alpha}_t) - \boldsymbol{\alpha}_t + \omega_t)$$

- $f(\mathbf{x}, \boldsymbol{\alpha})$ is contractive in $\mathbf{x}$:

$$\|f(\mathbf{x}_1, \boldsymbol{\alpha}) - f(\mathbf{x}_2, \boldsymbol{\alpha})\| \leq \lambda \|\mathbf{x}_1 - \mathbf{x}_2\|,$$

for some $0 \leq \lambda < 1$

- Unique fixed point for faster time-scale for given $\boldsymbol{\alpha}$ - the NE $\mathbf{x}^*(\boldsymbol{\alpha})$
- $g(\boldsymbol{\alpha})$ is non-expansive:

$$\|g(\boldsymbol{\alpha}_1) - g(\boldsymbol{\alpha}_2)\| \leq \|\boldsymbol{\alpha}_1 - \boldsymbol{\alpha}_2\|$$

- Two-time-scale SA widely studied when both time-scales have contractive mapping
- We have contractive in faster and non-expansive in slower time-scale
 - Leads to a slower decay rate

An interesting observation

- Why do we have to deal with a non-expansive mapping in the slower time-scale?
- Projection in the faster time-scale
 - Each player has a convex and compact action set
- In the absence of this projection, both time-scales have contractive mapping [a]
 - A rate of $\mathcal{O}(1/t)$ can be achieved [b]

^a S. Chandak, "Non-Expansive Mappings in Two-Time-Scale Stochastic Approximation: Finite-Time Analysis," *SIAM Journal on Control and Optimization* (2026).

^b S. Chandak, " $\mathcal{O}(1/k)$ Finite-Time Bound for Non-Linear Two-Time-Scale Stochastic Approximation," *IEEE Transactions on Automatic Control* (2026).

Conclusions

- Steering players towards a desirable equilibrium
- Discussed one specific scenario: Linear shift in utility
- Many other interesting scenarios with varying applications
 - Discrete control choices
 - Limited communication settings
- The intuition of time-scale separation often helps

Thank You!