

Learning to Control Unknown Multi-Agent Systems

Siddharth Chandak

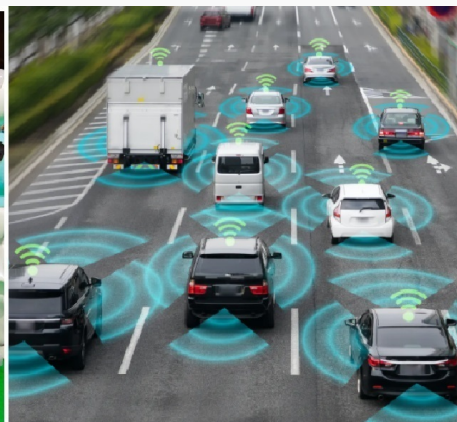
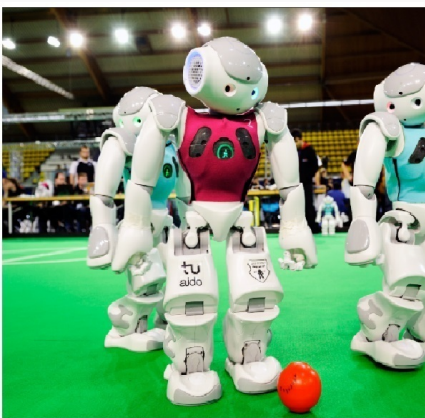
Date: August 6, 2026

Outline

- Overview
 - Game Control
 - Strongly Monotone Games and Nash Equilibrium
- Scenario I - Controllable Linear Coefficients
 - Two-time-scale Stochastic Approximation
- Scenario II - Discrete Game Parameters
 - Equilibrium Bandits
- Summary

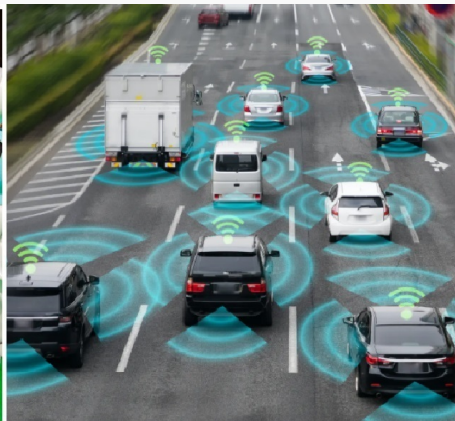
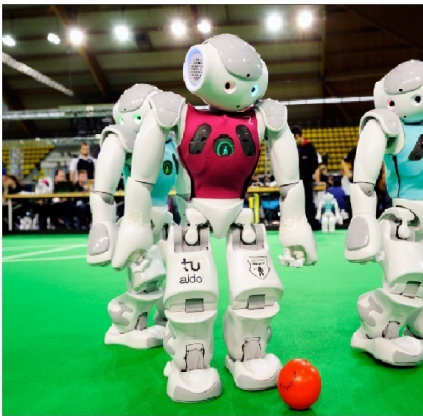
Overview

Multi-Agent Systems



Multi-Agent Games

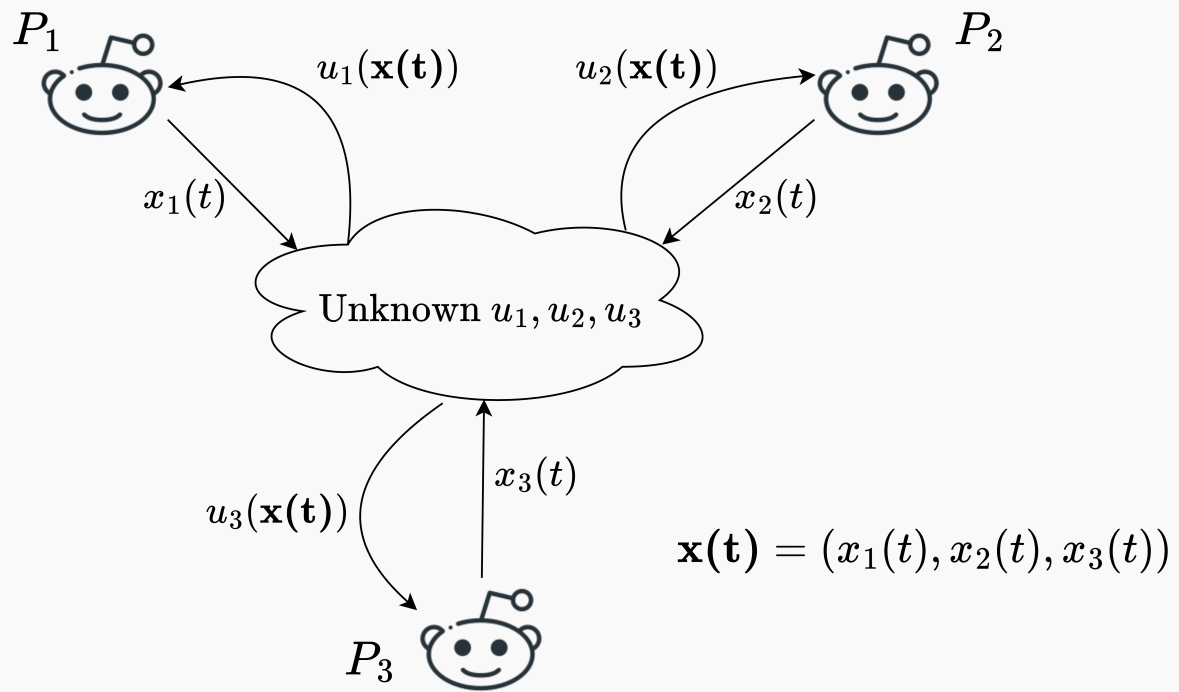
- Game with N agents or players
- Each player n takes action x_n
- Utility (Reward): $u_n(x_1, \dots, x_N)$



Local Objective

- **Local Objective:** Each player n wants to maximize their reward $u_n(x_1, \dots, x_N)$ under the constraint of limited feedback

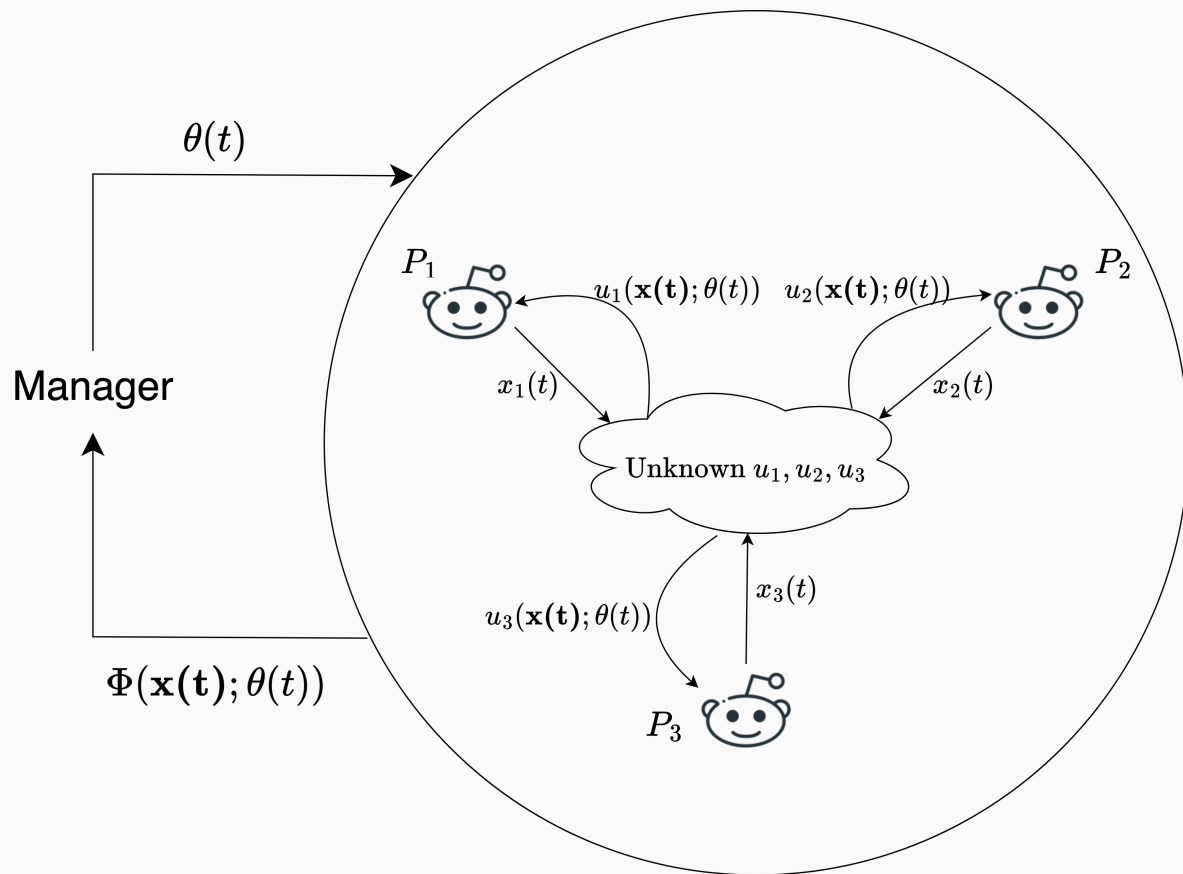
Bandit Feedback



Game Manager

- 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
 - We focus on the latter
- 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 $\mathbf{x}, \mathbf{y} \in \mathcal{X}$,

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

Nash Equilibrium

- 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

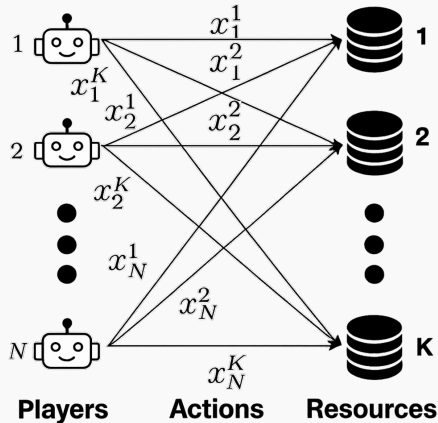
An action profile $\mathbf{x}^*$ is a pure Nash equilibrium (NE) if $u_n(x_n^*, \mathbf{x}_{-n}^*) \geq u_n(x_n, \mathbf{x}_{-n}^*)$, for all $x_n \in \mathcal{X}_n$ and all $n \in \mathcal{N}$.

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 $\mathbf{x}^*(\theta)$

Learning to Control Unknown Multi-Agent Systems

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

Scenario I: Controllable Linear Coefficients

S. Chandak, I. Bistritz, and N. Bambos, "Learning to Control Unknown Strongly Monotone Games," *IEEE Transactions on Control of Network Systems* (2026).

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 $\beta_t \in \mathbb{R}^{Nd}$, 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

Time-scale Separation

- Condition on stepsizes:

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

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

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

Results

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

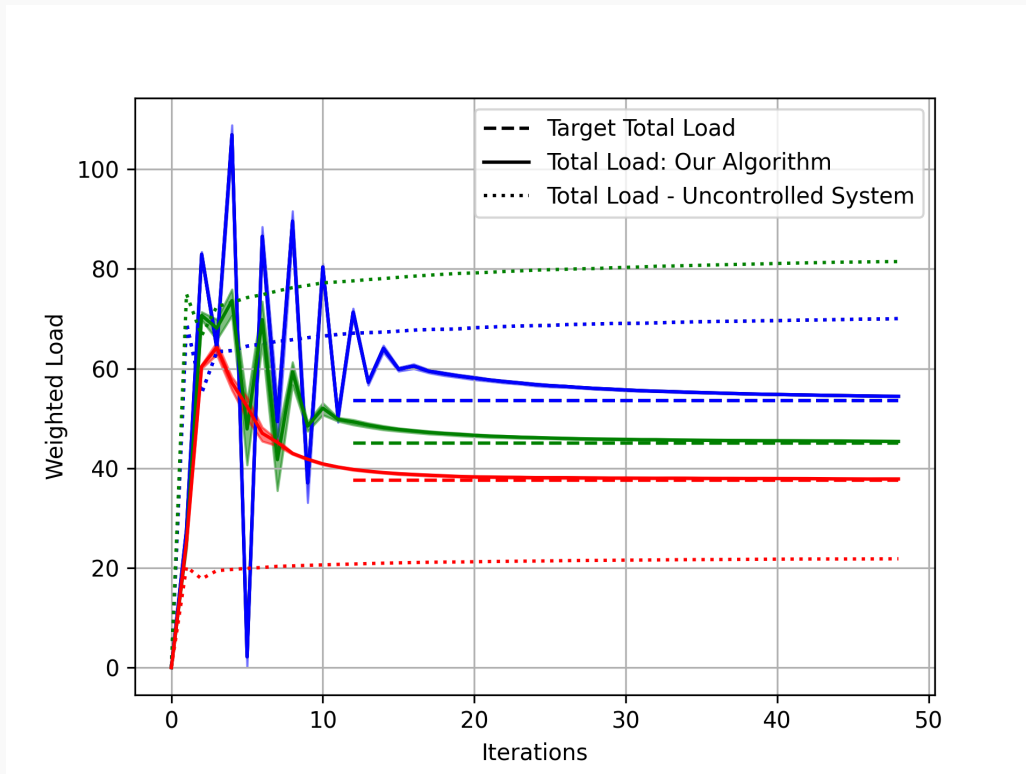
$$\lim_{t \rightarrow \infty} A\mathbf{x}_t = \ell^*,$$

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.

Analysis

$$\begin{aligned}\mathbf{x}_{t+1} &= \Pi_{\mathcal{X}} \left(\mathbf{x}_t + \eta_t (F(\mathbf{x}_t) - A^T \alpha_t + M_{t+1}) \right) \\ \alpha_{t+1} &= \alpha_t + \epsilon_t (A\mathbf{x}_t - \ell^*)\end{aligned}$$

- Can be expressed as fixed-point iterations:

$$\begin{aligned}\mathbf{x}_{t+1} &= \Pi_{\mathcal{X}} (\mathbf{x}_t + \eta_t (f(\mathbf{x}_t, \alpha_t) - \mathbf{x}_t + M_{t+1})) \\ \alpha_{t+1} &= \alpha_t + \epsilon_t (g(\alpha_t) - \alpha_t + \omega_t)\end{aligned}$$

- 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

Analysis

$$\begin{aligned}\mathbf{x}_{t+1} &= \Pi_{\mathcal{X}}(\mathbf{x}_t + \eta_t(f(\mathbf{x}_t, \boldsymbol{\alpha}_t) - \mathbf{x}_t + M_{t+1})) \\ \boldsymbol{\alpha}_{t+1} &= \boldsymbol{\alpha}_t + \epsilon_t(g(\boldsymbol{\alpha}_t) - \boldsymbol{\alpha}_t + \omega_t)\end{aligned}$$

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

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

Analysis

- 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
 - Requires novel analysis
 - 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 [Chandak (2026a)]
 - A rate of $\mathcal{O}(1/t)$ can be achieved [Chandak (2026b)]

^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).

Beyond our discussion

- Other applications:
 - Weighted resource allocation
 - Can also model settings where the manager wants to minimize a global quadratic cost
- Our algorithm can also be used when the target linear constraints are changing over time
- Using the same two-time-scale intuition:
 - Sparse communication by manager but still achieve same theoretical guarantees

Scenario II: Discrete Game Parameters

S. Chandak, I. Bistriz, and N. Bambos, "Equilibrium Bandits: Learning Optimal Equilibria of Unknown Dynamics," *Autonomous Agents and Multiagent Systems (AAMAS)* (2023).

Problem Formulation

- Manager has to choose from a discrete set of parameters $\theta \in \{1, \dots, K\}$
 - Can be thought of as K different policies
- Maximize global objective $\Phi(\mathbf{x}; \theta)$
- Example: Resource Allocation
 - Manager decides which subset of resources each player can use
 - Each $\theta \in \{1, \dots, K\}$ denote this subset for each player
 - Under action θ , player n has only access to resources $\mathcal{R}_n(\theta) \subseteq \{1, \dots, d\}$

Practical Implementation

- Odd-Even rationing rule



Problem Formulation

- Manager chooses θ_t at $t = 0, \dots$,
- Players update their action using gradient ascent:

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

- Manager observes noisy global reward $y_t = \Phi(\mathbf{x}_t; \theta_t) + M_{t+1}$

Formulating the Manager's Objective

- Manager cares about the objective at Nash equilibrium
- Optimal policy defined with respect to global objective at corresponding NE:

$$\theta^* = \arg \max_{\theta} \Phi(\mathbf{x}^*(\theta); \theta)$$

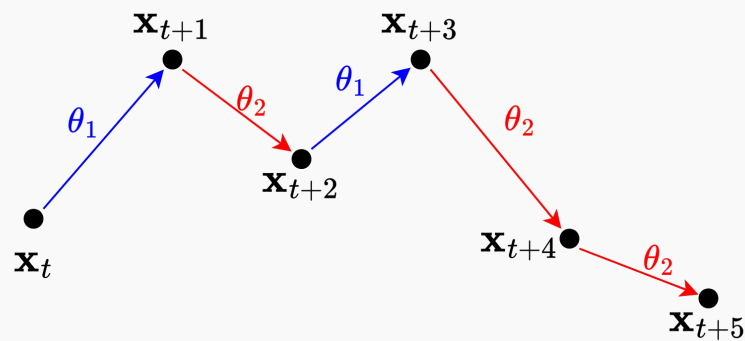
- Regret:

$$\mathbb{E}[R(T)] = \mathbb{E} \left[\sum_{t=1}^T (\Phi(\mathbf{x}^*(\theta^*); \theta^*) - \Phi(\mathbf{x}_t; \theta_t)) \right]$$

- Defined w.r.t. what the optimal policy achieves at equilibrium
- Incentivize the manager to choose the optimal policy and allow the players to converge quickly

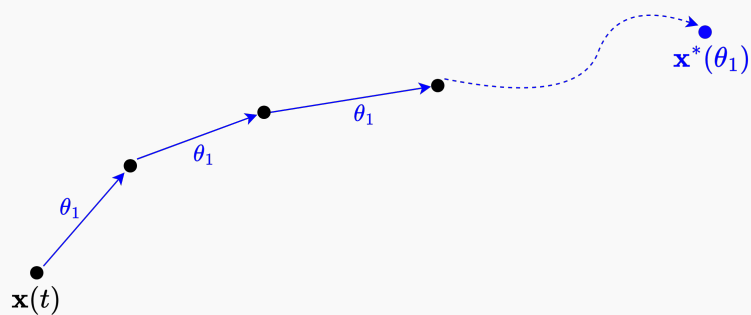
How is this different?

- Cannot switch policy at every step
 - Unlike the previous scenario where the control parameter could be changed continuously, we have discrete choices here
 - Would learn nothing about the objective at NE



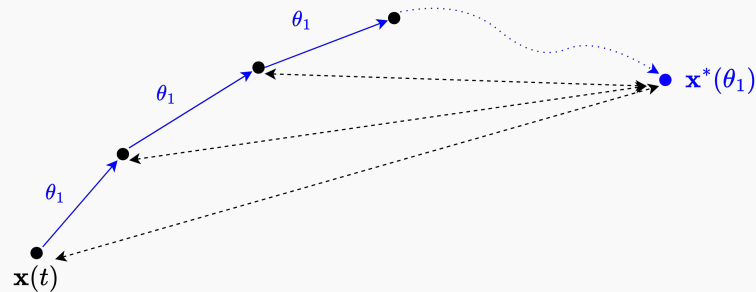
Distance from NE

- Converges to NE if policy is fixed



- But how long to wait for convergence?

Distance from NE



- Distance from NE $\mathbf{x}^*(\theta)$ decreases when policy θ is implemented, i.e.,

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*(\theta)\| \leq \exp\left(-\frac{1}{\tau_c}\right) \|\mathbf{x}_t - \mathbf{x}^*(\theta)\|,$$

where policy at time t is θ

Distance from NE

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*(\theta)\| \leq \exp\left(-\frac{1}{\tau_c}\right) \|\mathbf{x}_t - \mathbf{x}^*(\theta)\|$$

- τ_c : *'approximate' convergence time to equilibrium*
- $\exp(-1/\tau_c) = \sqrt{1 - 2\mu\eta + L_G^2\eta^2}$
 - μ : strongly monotone parameter of game
 - L_G : Lipschitz constant for concatenated gradient operator $G(\cdot)$
 - η : Stepsize used by players for gradient ascent

Equilibrium Bandits

- Model this problem as a modification of the stochastic multi-armed bandit problem
 - Each policy is an arm
 - The exact true reward (+ stochastic noise) of an arm is known only after playing it infinitely often
- Solve this problem using optimism-based algorithm
 - Modification of Upper Confidence Bound
 - **Upper Equilibrium Confidence Bound**
 - Three major additions

The Key Idea: Bounds on Objective at NE

- Want to determine how the players will behave at equilibrium for a policy without waiting for convergence
 - Recall: Distance from NE $\mathbf{x}^*(\theta)$ decreases when policy θ is implemented, i.e.,

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*(\theta)\| \leq \exp\left(-\frac{1}{\tau_c}\right) \|\mathbf{x}_t - \mathbf{x}^*(\theta)\|,$$

where policy at time t is θ

- **Approach:** Can use this to get a bound on the global objective at NE for a policy
 - Suppose policy θ is chosen consecutively ℓ times (from t to $t + \ell$):

$$\Phi(\mathbf{x}_{t+\ell}; \theta) - Le^{-\frac{\ell}{\tau_c}} \leq \Phi(\mathbf{x}^*(\theta); \theta) \leq \Phi(\mathbf{x}_{t+\ell}; \theta) + Le^{-\frac{\ell}{\tau_c}},$$

where L is Lipschitz constant for $\Phi(\cdot; \theta)$.

Modification II - Epochs of Increasing Length

- Need to keep policy fixed for a consecutive number of times
- **Approach:** Epoch-based system: policies are changed only at ends of epochs
- Lengths of epochs increased as a policy is chosen more times
 - *Intuition:* Promising policies are given more time to converge
 - If policy θ has been chosen for m epochs, then length of $(m + 1)^{th}$ epoch is e^{m+1} time-steps

Modification III: Noise Averaging

- Manager observes noisy global objective: need to average to eliminate noise
- Cannot average all rewards from an epoch (or from older epochs):
 - Far from equilibrium, hence less information about reward at equilibrium
- **Approach:** If policy θ is implemented for ℓ consecutive steps in an epoch, take average of last $\ell/2$ observed rewards

UECB: Bring it Together

Algorithm (UECB)

For epoch $n = 1, 2, \dots$

(1) Implement policy $\theta_n = \arg \max_{\theta} \text{UECB}_{\theta}$ for $\ell_n = \exp(m_{\theta_n} + 1)$ time-steps

(2) Estimate:

$$\hat{\Phi}_{\theta,n} = \frac{1}{\ell_n/2} \sum_{t=t_n+\ell_n/2}^{t_n+\ell_n} y_t$$

(3) Update UECB:

$$\text{UECB}_{\theta,n} = \hat{\Phi}_{\theta,n} + \frac{c_1}{\ell_n/2} \exp\left(-\frac{\ell_n}{2\tau_c}\right) + \sqrt{\frac{c_2\sigma^2}{\ell_n/2} \log(2t_n^3)}$$

End

UECB: Bring it Together

Algorithm (UECB)

For epoch $n = 1, 2, \dots$

(1) Implement policy $\theta_n = \arg \max_{\theta} \text{UECB}_{\theta}$ for $\ell_n = \exp(m_{\theta_n} + 1)$ time-steps

(2) Estimate:

$$\hat{\Phi}_{\theta,n} = \frac{1}{\ell_n/2} \sum_{t=t_n+\ell_n/2}^{t_n+\ell_n} y_t$$

(3) Update UECB:

$$\text{UECB}_{\theta,n} = \hat{\Phi}_{\theta,n} + \underbrace{\frac{c_1}{\ell_n/2} \exp\left(-\frac{\ell_n}{2\tau_c}\right)}_{\text{Equilibrium Bias}} + \underbrace{\sqrt{\frac{c_2\sigma^2}{\ell_n/2} \log(2t_n^3)}}_{\text{Noise Averaging } (\sim \text{UCB})}$$

End

Results

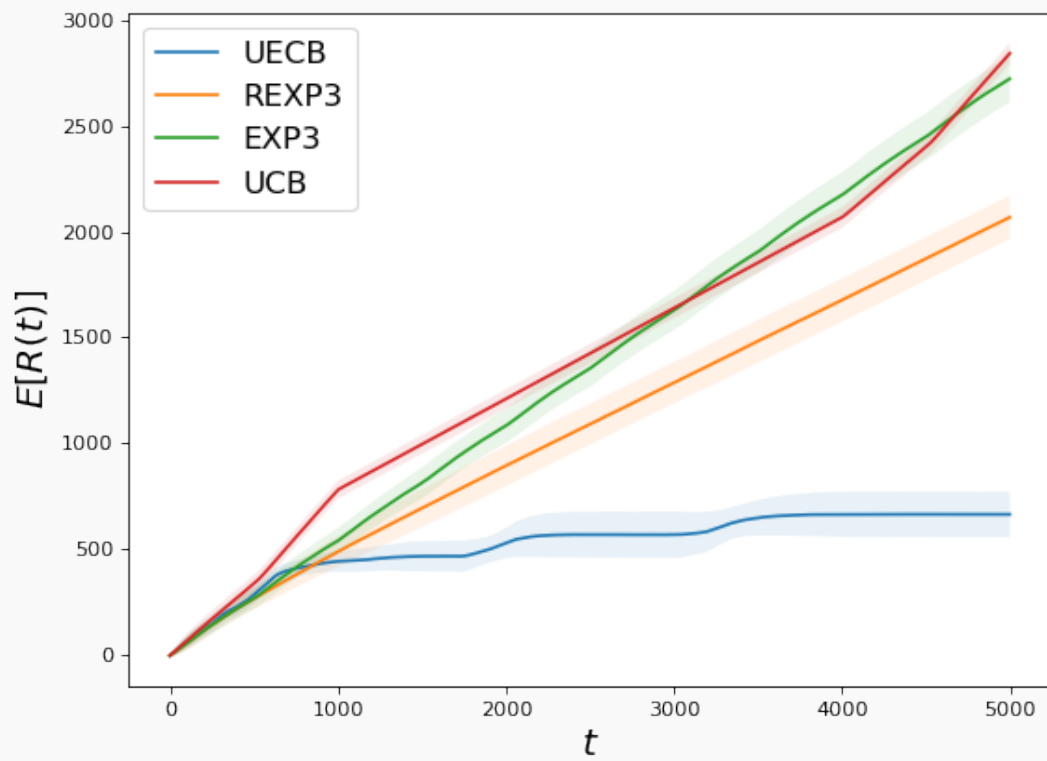
Theorem

The regret achieved by UECB algorithm is bounded as:

$$\mathbb{E}[R(T)] = \mathcal{O} \left(\sum_{\theta \neq \theta^*} \frac{\log(T)}{\Delta_\theta} + \tau_c \log \left(\tau_c \log \left(\frac{1}{\Delta_\theta} \right) \right) + \tau_c \log(\log(T)) \right)$$

where Δ_θ is the suboptimality gap for policy θ defined w.r.t. equilibrium rewards.

Simulations



Beyond our discussion

- Another interesting application: **Epidemic control**
 - How should the government control a new epidemic?
 - Multiple policies:
 - e.g., lockdown, mask enforcement, advertising for awareness
 - Each has their own operational cost
 - Affect the spread differently
 - Care about the equilibrium infection rate for each policy

Summary

Summary

- Game control under two different scenarios
- Scenario I: Controllable linear coefficients
 - Intuition: pricing and subsidies
 - Proposed a two-time-scale method for convergence to desirable NE
- Scenario II: Discrete game control parameters
 - Intuition: different policies
 - Developed UECB, an optimism-based bandit algorithm

Summary

- Many other scenarios with varying assumptions and applications:
 - Beyond strongly monotone games?
 - Beyond Nash equilibrium?
- Automation is getting more prevalent
 - Need to model interactions between humans and AI better
 - Designing such models (and algorithms for them) require new tools and rethinking old ones

Publications: Multi-Agent Systems & RPM

Multi-Agent Systems

- S. Chandak, R. Tamizholi, and N. Bambos, "Last-Iterate Guarantees for Learning in Co-coercive Games," *IEEE Conference on Decision and Control (CDC)*, 2026 (accepted).
- S. Chandak, I. Bistritz, and N. Bambos, "Choose Your Battles: Distributed Learning Over Multiple Tug of War Games," *IEEE Transactions on Automatic Control*, 2026.
- S. Chandak, I. Bistritz, and N. Bambos, "Learning to Control Unknown Strongly Monotone Games," *IEEE Transactions on Control of Network Systems*, 2026.
- S. Chandak, I. Bistritz, and N. Bambos, "Tug of Peace: Distributed Learning for Quality of Service Guarantees," *IEEE Conference on Decision and Control (CDC)*, 2023.
- S. Chandak, I. Bistritz, and N. Bambos, "Equilibrium Bandits: Learning Optimal Equilibria of Unknown Dynamics," *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2023.
- S. U. Haque, S. Chandak, F. Chiariotti, D. Günduz, and P. Popovski, "Learning to Speak on Behalf of a Group: Medium Access Control for Sending a Shared Message," *IEEE Communications Letters*, 2022.

Remote Patient Monitoring

- R. Tamizholi, S. Chandak, I. Thapa, N. Bambos and D. Scheinker, "Safe Learning of Adaptive Control Policies for Remote Patient Monitoring," *IEEE Healthcom*, 2026 (submitted).
- S. Chandak, I. Thapa, N. Bambos and D. Scheinker, "Optimal Control for Remote Patient Monitoring with Multidimensional Health States", *IEEE International Conference on Communications (ICC)*, 2025.
- S. Chandak, I. Thapa, N. Bambos and D. Scheinker, "Tiered Service Architecture for Remote Patient Monitoring", *IEEE Healthcom*, 2024.

Publications: SA, RL, & Optimization

- S. Chandak, " $O(1/k)$ Finite-Time Bound for Non-Linear Two-Time-Scale Stochastic Approximation," *IEEE Transactions on Automatic Control*, 2026.
- S. Chandak, "Non-Expansive Mappings in Two-Time-Scale Stochastic Approximation: Finite-Time Analysis," *SIAM Journal on Control and Optimization*, 2026 (accepted).
- S. Chandak and V. S. Borkar, "A Concentration Bound for TD(0) with Function Approximation," *Stochastic Systems*, 2026.
- S. Chandak, S. U. Haque, and N. Bambos, "Finite-Time Bounds for Two-Time-Scale Stochastic Approximation with Arbitrary Norm Contractions and Markovian Noise," *IEEE Conference on Decision and Control (CDC)*, 2025.
- S. Chandak, P. Shah, V. S. Borkar, and P. Dodhia, "Reinforcement Learning in Non-Markovian Environments," *Systems and Control Letters*, 2024.
- S. Chandak, V. S. Borkar, and H. Dolhare, "A Concentration Bound for LSPE(λ)," *Systems and Control Letters*, 2023.
- S. Chandak, V. S. Borkar, and P. Dodhia, "Concentration of Contractive Stochastic Approximation and Reinforcement Learning," *Stochastic Systems*, 2022.

Submitted Articles

- S. Chandak, "Concentration and Mean-Square Bounds for Contractive Stochastic Approximation: A Unified Elementary Approach," *Mathematics of Control, Signals, and Systems*.
- A. Kar, S. Chandak, R. Singh, S. Sinhaajari, E. Moulines, S. Bhatnagar, and N. Bambos, "Policy Gradient Methods for Non-Markovian Reinforcement Learning," *NeurIPS*, 2026.
- R. Singh, S. Chandak, V. S. Borkar, E. Moulines, and N. Bambos, "Regret and Sample Complexity of Online Q-Learning via Concentration of Stochastic Approximation with Time-Inhomogeneous Markov Chains," *NeurIPS*, 2026.
- S. Chandak, A. Yadav, A. Ozgur, and N. Bambos, "Heavy-Tailed and Long-Range Dependent Noise in Stochastic Approximation: A Finite-Time Analysis," *IEEE Transactions on Automatic Control*.
- A. Kar, S. Chandak, R. Singh, E. Moulines, S. Bhatnagar, and N. Bambos, "High-Probability Bounds for SGD under the Polyak-Łojasiewicz Condition with Markovian Noise," *SIAM Journal on Optimization*

Thank You!