Equilibrium Bandits: Learning Optimal Equilibria of Unknown Dynamics

Siddharth Chandak, Ilai Bistritz, Nicholas Bambos

June 1, 2023

AAMAS 2023

Outline

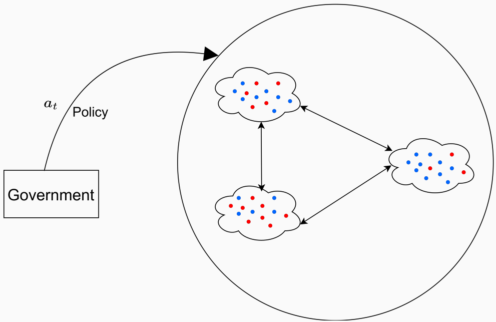
- Motivation: Equilibrium Bandits
- UECB Algorithm
- Results

Motivation: Equilibrium Bandits

Application: Epidemic Control

- How should the government control a new epidemic?
- Hard to model the epidemic and population interaction
- Multiple policies:
  - e.g., lockdown, mask enforcement, advertising for awareness
  - Each has their own operational cost
  - Affect the spread of epidemic differently
- Care about the equilibrium infection rate of each policy:
  - Need to enact it consecutively for a "large number of time-steps"

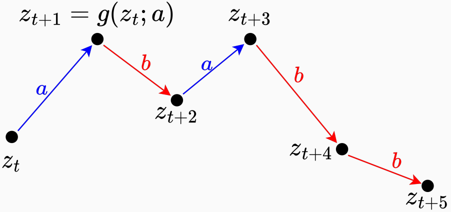
Application: Epidemic Control



- Multiple policies:
  - e.g., lockdown, mask enforcement, advertising for awareness
  - Each has their own operational cost
  - Affect the spread of epidemic differently
- Care about the equilibrium infection rate of each policy:
  - Need to enact it consecutively for a "large number of time-steps"

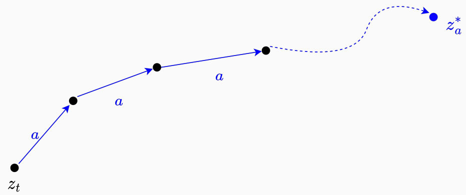
Equilibrium Bandits: Problem Formulation

- Agent takes action $a_t \in \{1, \dots, K\}$ at each time $t = 0, 1, 2, \dots$
- $\overrightarrow{z_t}$: System State
  - Evolution Function: $z_{t+1} = g(z_t; a_t)$



Equilibrium Bandits: Problem Formulation

- $\overrightarrow{z_t}$: System State
  - Evolution Function: $z_{t+1} = g(z_t; a_t)$
  - Each action a has their equilibrium point z_a^*
  - Converges if action is fixed, i.e., $\lim_{t\uparrow\infty} g^{(t)}(z; a) = z_a^*$

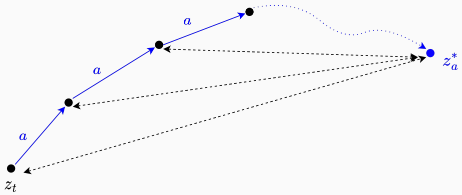


Equilibrium Bandits: Problem Formulation

- $\overrightarrow{z_t}$: System State
  - Distance from equilibrium decreases when action a is played, i.e.,

$$\|g(z,a) - z_a^*\| \leq \exp\left(-\frac{1}{\tau_c}\right)\|z - z_a^*\|$$

  - τ_c: *approximate convergence time to equilibrium*



Equilibrium Bandits: Problem Formulation

- $f(z_t; a_t)$: Reward Function
- Agent receives noisy rewards
- Optimal action a^*: action with maximum reward at equilibrium

$$a^* = \arg\max_a f(z_a^*, a)$$

- Regret:

$$\mathbb{E}[R(T)] = \mathbb{E}\left[\sum_{t=1}^{T}(f(z_{a^*}^*; a^*) - f(z_t; a_t))\right] \tag{1}$$

  - Difference w.r.t. what the optimal action achieves at equilibrium

Equilibrium Bandits: Problem Formulation

- $f(z_t; a_t)$: Reward Function
- Agent receives noisy rewards y_t
- Optimal action a^*: action with maximum reward at equilibrium

$$a^* = \arg\max_a f(z_a^*, a)$$

- Regret:

$$\mathbb{E}[R(T)] = \mathbb{E}\left[\sum_{t=1}^{T}(f(z_{a^*}^*; a^*) - f(z_t; a_t))\right] \tag{2}$$

  - Difference w.r.t. what the optimal action achieves at equilibrium
  - Want to incentivize choosing the optimal arm and converging quickly

Application: Epidemic Control

- **Agent**: Government
- **Actions**: Policies
- **System State** (z_t): Infection Rate in Population
- **Evolution Function** $(g(z_t; a_t))$: Spread of epidemic
- **Reward Function** $(f(z_t; a_t))$: Negative Cost
  - Cost due to infection
  - Operational cost
- **Regret**: How we perform as compared to the optimal policy

Upper Equilibrium Concentration Bound (UECB)

Challenges

- Cannot switch action at every time-step
  - Would learn nothing about the reward at equilibrium
- Cannot wait too long
  - Can be very costly, e.g., epidemic
  - Would need to know τ_c and suboptimality gap to determine how long to wait

UECB: Convergence Bounds

- Want to determine how an action will behave at equilibrium without waiting for convergence
  - Recall: Distance from equilibrium decreases when action a is played,

$$\|g(z,a) - z_a^*\| \le \exp\left(-\frac{1}{\tau_c}\right)\|z - z_a^*\|$$

- **Approach:** Can use this to get a bound on how well an action can perform at equilibrium
  - Suppose action a is player consecutively ℓ times (from t to $t+\ell$):

$$f(a; z_{t+\ell}) - Le^{-\frac{\ell}{\tau_c}} \le f(a; z_a^*) \le f(a; z_{t+\ell}) + Le^{-\frac{\ell}{\tau_c}}$$

UECB: Epochs of Increasing Length

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

UECB: Noise Averaging

- Receive noisy rewards: 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 action a is played for ℓ consecutive steps in an epoch, take average of last $\ell/2$ observed rewards

UECB: Bring it Together

Algorithm (UECB)

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

(1) Play action $a_n = \arg\max_a \text{UECB}_a$ for $\ell_n = \exp(m_a + 1)$ time-steps

(2) Estimate:

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

(3) Update UECB:

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

End

UECB: Bring it Together

Algorithm (UECB)

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

(1) Play action $a_n = \arg\max_a \text{UECB}_a$ for $\ell_n = \exp(m_a + 1)$ time-steps

(2) Estimate:

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

(3) Update UECB:

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

End

Results

Guarantees

Theorem

For any instance of equilibrium bandits, the regret achieved by UECB algorithm is bounded as:

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

where Δ_a is the suboptimality gap for arm a defined w.r.t. equilibrium rewards.

Guarantees: What does each term mean?

Theorem

For any instance of equilibrium bandits, the regret achieved by UECB algorithm is bounded as:

$$\mathbb{E}[R(T)] = \mathcal{O}\left(\sum_{a \neq a^*} \underbrace{\frac{\log(T)}{\Delta_a}}_{\substack{\text{Stochastic}\\ \text{Bandits}}} + \underbrace{\tau_c \log\left(\tau_c \log\left(\frac{1}{\Delta_a}\right)\right) + \tau_c \log(\log(T))}_{\text{Convergence Time}} \right)$$

where Δ_a is the suboptimality gap for arm a defined w.r.t. equilibrium rewards.

- τ_c: *Approximate convergence time to equilibrium*

Lower Bound

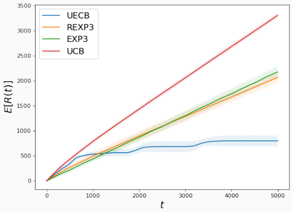
Theorem

There exist instances of equilibrium bandits, where for all 'good' algorithms

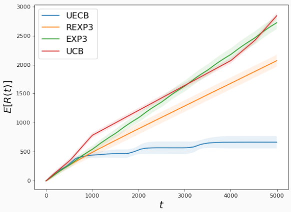
$$\mathbb{E}[R(T)] = \Omega\left(\frac{\log(T)}{\Delta_a} + \tau_c \Delta_a \log\left(\frac{1}{\Delta_a}\right)\right).$$

- UECB is optimal in T, Δ_a, and optimal upto logarithmic factors in τ_c
- Lower bound obtained using an instance where arms cannot be distinguished for the first $\sim\tau_c$ steps

Numerical Experiments



(a) SIS Epidemic Control



(b) Strongly Monotone Games

- Strongly Monotone Games:
  - Game designer tries to optimize global objective by controlling game parameters
  - Players optimize local utility using gradient ascent
  - Given fixed parameters, players slowly converge to Nash equilibrium
- UECB obtains logarithmic regret while standard algorithms such as UCB and EXP3 achieve linear regret

Summary

- Equilibrium Bandits: A new bandit problem
  - Can be used to make optimal decisions for complex systems which slowly evolve and converge to some equilibrium
  - Examples include epidemic control, game control, congestion control
- UECB Algorithm:
  - Inspiration from UCB
  - Concept of Convergence Bounds

Thank You!