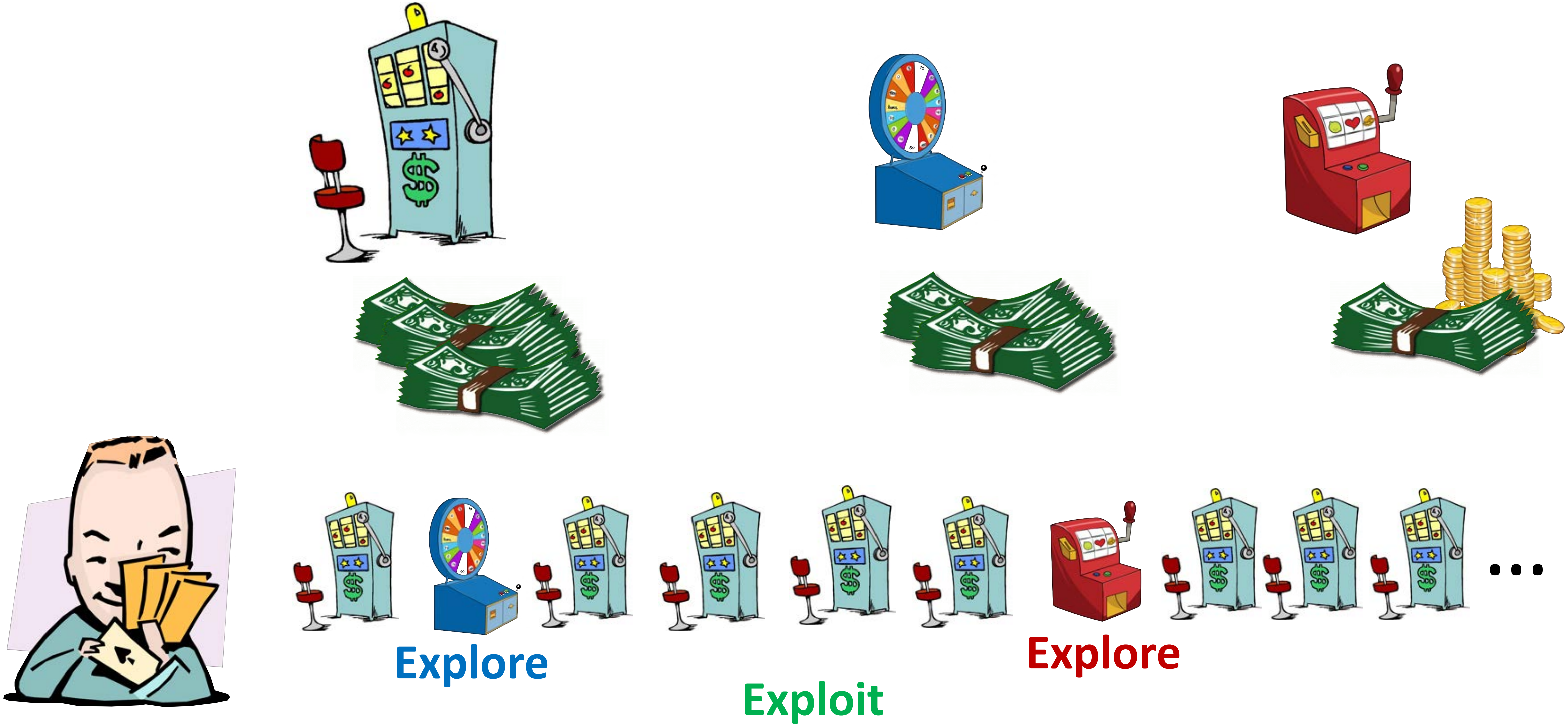

# Blocking Bandits

Soumya Basu[1]

joint with Rajat Sen[2], Sujay Sanghavi[1,2], Sanjay Shakkottai[1]
[1]UT Austin, [2]Amazon

# Multi Armed Bandit



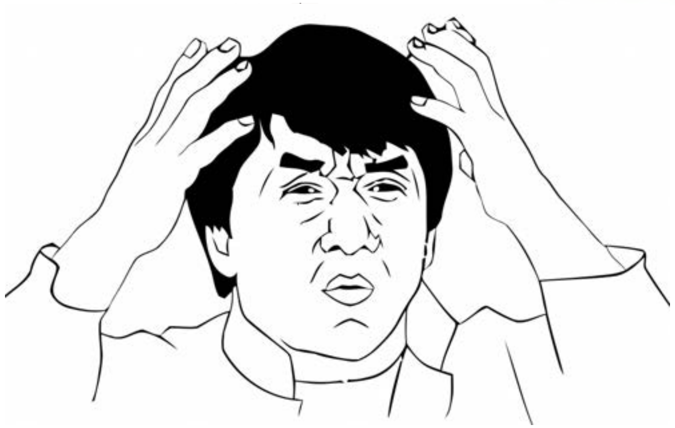

All images/clip arts are taken from open sourced websites

# Blocking Bandits


2 months


1 month


2 months



**How do you game the system?**

All images/clip arts are taken from open sourced websites

# Outline

- Blocking Bandits Model
- Applications
- Offline Optimization
- Online Learning
- Future Directions
- Research Overview

# Blocking Bandits Model

| | | | | |
|---|---|---|---|---|
| Arms: | 1 | 2 | … | K |
| Mean Rewards: | $\mu_1$ | $\mu_2$ | … | $\mu_K$ |
| Fixed Delays: | $D_1$ | $D_2$ | … | $D_K$ |

$\mu_i$ unknown
$D_i$ known

Each time arm $i$ is played, arm $i$ is **blocked** for the next $(\boldsymbol{D_i - 1})$ time steps

**Objective:** Maximize the expected reward in T time slots

$Unit\ Delay: \forall i, D_i = 1 \equiv$ Multi armed bandit problem

# Applications: Job scheduling with Maximum QoS

- Arms are **servers/machines**
- Each timeslot one task arrives
- Server $i$ has processing time $D_i$ (Service time varies across servers)
- Server $i$ provide quality of service (QoS) $\mu_i$
- Tasks are **homogeneous**
  $\rightarrow$ Identical QoS distribution, and processing time for individual user

# Applications: Ad Placement with Gap Constraint

- Arms are **users/subscribers**
- Each timeslot one ad needs to be placed
- User $i$ has a gap constraint of $D_i$ (Avoid annoyance)
- User $i$ has a mean click through rate (CTR) of $\mu_i$
- Ads are **homogeneous**
  $\rightarrow$ Identical CTR  distribution and gap for individual user

**Other applications:**
- Homogeneous Product recommendation
- Point to point shuttle service

# Off-the-Shelf Solutions

- **Combinatorial Semi-Bandits**
  - Take decisions for a block of time and observe all rewards in each block
  - Approaches [Y. Gai et al. 12, B. Kveton et al. 14, …]
  - Block length = $lcm(\{D_i: i = 1 \; to \; K\})$

**Existing Methods are Computationally Intractable!**

- **Online Markov Decision Processes**
  - Markov chain with known transition and unknown stochastic reward
  - Approaches [P. Auer et al. 07, A. Tewari et al. 08, G. Neu et al. 09, A Zimin et al. 13,…]
  - State Space = $\prod_{i \in [K]} D_i$, Horizon = $lcm(\{D_i: i = 1 \; to \; K\})$

# Offline Optimization Problem: Formulation

- The mean rewards of the arms $(\boldsymbol{\mu_i})$ are known
- $\boldsymbol{a_t}$: Selected arm at time $t$
- **Blocking Constraint**:

$$\forall \boldsymbol{i}, \boldsymbol{min}\{|\boldsymbol{t}-\boldsymbol{t}'|: \boldsymbol{t}, \boldsymbol{t}' \leq \boldsymbol{T}, \boldsymbol{a_t} = \boldsymbol{a_{t'}} = \boldsymbol{i}\} \geq \boldsymbol{D_i} \ \ (*)$$

- **Optimal Expected Reward:** $\mathrm{OPT} = \max_{\substack{\{a_t:\, t \leq T\} \\ s.t. (*)\ holds}} \sum_{t=1}^{T} \mu_{a_t}$

Combinatorial optimization problem across timeslots

# Offline Optimization Problem: Hardness

- **Optimal Expected Reward:** $$\mathrm{OPT} = \max_{\substack{\{a_t:\, t \le T\} \\ s.t. (*)\ holds}} \sum_{t=1}^{T} \mu_{a_t}$$

Computationally as “Hard” as Dense PINWHEEL Scheduling **Result 1**

“Hard”: NO pseudo-polynomial time algorithm under randomized Exponential Time Hypothesis

# Offline Optimization Problem: Approximation

- **Example 1:** Greedy-Reward vs Optimal

| Arm | $\mu_i$ | $D_i$ |
|---|---|---|
| 1 | 1 | 4 |
| 2 | 1 | 4 |
| 3 | 0.9 | 2 |
| 4 | 0.1 | 1 |

**Only Arm 4 available**

| Greedy | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|---|---|---|

**Reward:** $3\, floor\left(\frac{T}{4}\right) + O(1)$

| Optimal | 3 | 1 | 3 | 2 | 3 | 1 | 3 | 2 |
|---|---|---|---|---|---|---|---|---|

**Reward:** $2.9\, floor\left(\frac{T}{3}\right) + O(1)$

**Order Matters**
**Start with 3**

**Greedy-(Reward/Delay)**

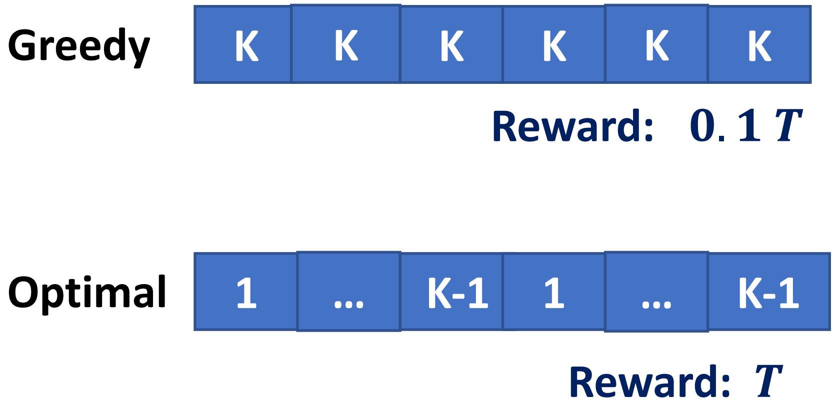
**There exists an instance where Greedy-Reward obtains 3/4 of the Optimal Reward**

Make reward of Arm 4 close to 0 and reward of Arm 3 close to 1

# Offline Optimization Problem: Approximation

- **Example 2:** Greedy-Reward/Delay vs Optimal

| Arm | $\mu_i$ | $D_i$ |
|---|---|---|
| 1 | 1 | K-1 |
| … | … | |
| K-1 | 1 | K-1 |
| K | 0.1 | 1 |




**Greedy-(Reward/Delay) is**
**Arbitrarily bad**

**There exists an instance where Greedy-(Reward/Delay) obtains O(1/K) of the Optimal Reward**

Make reward of Arm K close to 1/K

# Offline Optimization Problem: Approximation

**Greedy-Reward obtains at least ((1-1/e) OPT – O(1)) reward** **Result 2**

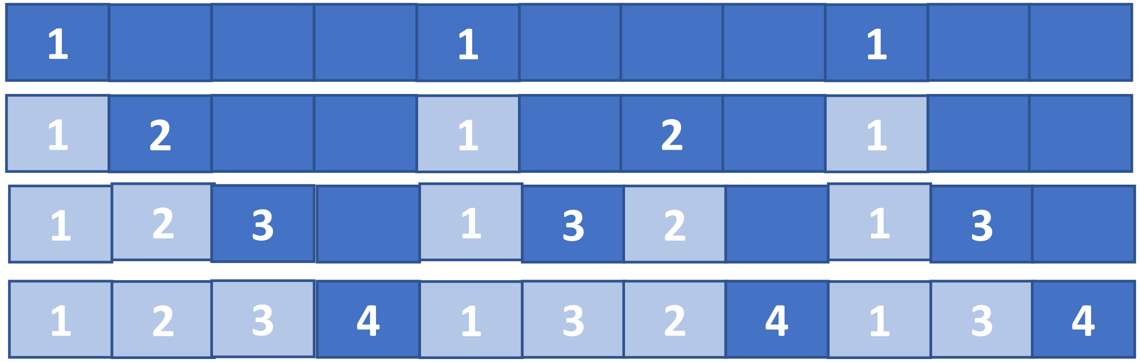
**OPT** = $\Theta(\mathrm{T})$

- **LP Based Upper Bound on OPT**:
  - Let the arms be sorted: $1 \geq \mu_1 \geq \mu_2 \geq \ \dots \geq \mu_K \geq 0$
  - Arm $i$ can be played **at most** $\boldsymbol{ceil(T/D_i)}$ many times
  - **LP**: $\max_{n_i} \sum_{i=1}^{K} n_i \mu_i \,,\ s.t.\, 0 \leq n_i \leq ceil\left(\frac{T}{D_i}\right) \ \forall i \in [K]$
  - Let $K^* = \min\{i : \sum_{j=1}^{i} 1/D_j \geq 1\}$

$$OPT \leq \sum_{i=1}^{K^*} \mu_i ceil(T/D_i)$$

# Offline Optimization Problem: Approximation

- Greedy-reward plays the best available arm in each time slot
- **Lower Bound on Greedy-Reward (Iterative Periodic):**
  - Periodically place the current best arm and delete used time slots




| Arm | $\mu_i$ | $D_i$ |
|---|---|---|
| 1 | 1 | 4 |
| 2 | 1 | 4 |
| 3 | 0.9 | 2 |
| 4 | 0.1 | 1 |

**Greedy-Reward**

# Offline Optimization Problem: Approximation

- **Lower Bound on Greedy-Reward (contd.):**
  - Arm $i$ can be used at least $\frac{\mathrm{T}}{\mathrm{D_i}}\prod_{\mathrm{j}=1}^{i-1}\left(1-\frac{1}{D_j}\right)$ - O(1) times (induction on i)

$$\text{Greedy-Reward} \geq \sum_{i=1}^{K} \boldsymbol{\mu_i} \frac{\boldsymbol{T}}{\boldsymbol{D_i}} \prod_{\mathbf{j}=\mathbf{1}}^{\boldsymbol{i}-\mathbf{1}} \left(\mathbf{1}-\frac{\mathbf{1}}{\boldsymbol{D_j}}\right) - O(1)$$

- **Approximation Guarantee:**
  - Lower bound: Min $\frac{Greedy\ Lower\ Bound}{LP\ Upper\ Bound}$ over $\mu_i, D_i$
  - Subject to: $1 \geq \mu_1 \geq \mu_2 \geq \ \ldots \geq \mu_K \geq 0$ and $D_i \geq 1\ \forall i$

# Online Learning: $\alpha$ Regret

- The mean rewards $\mu_i$ are unknown
- How learning affects the reward?

$$\alpha \text{ Regret} = \alpha\ OPT\ - \mathbb{E}[\sum_{t=1}^{T} \mu_{a_t}]$$

- Regret notion used in combinatorial bandits
  [V. Dani et al. 2008, W Chen et al. 2013, …]

- O(log(T)) regret w.r.t. Greedy-Reward ≡ O(log(T)) $(1 - 1/e)Regret$

# Online Learning: Greedy-UCB-Reward

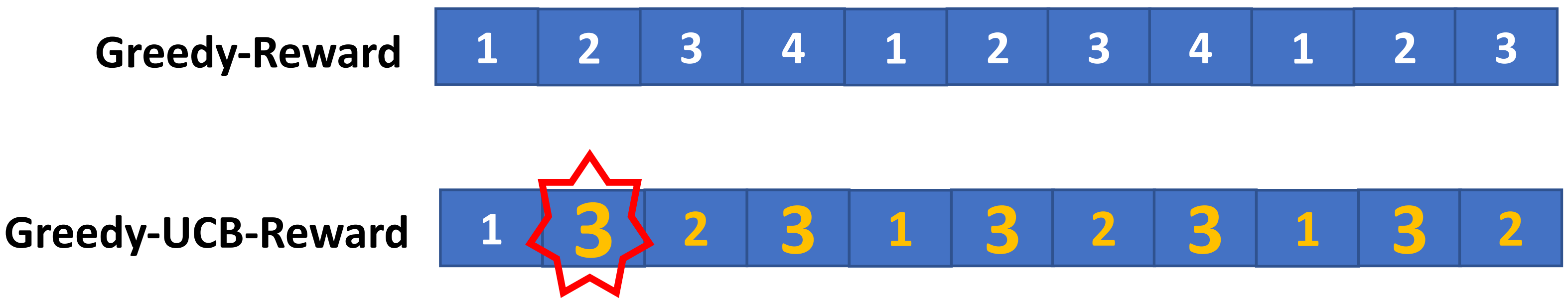
- $N_i(t)$: Number of times arm $i$ played upto time t
- $\widehat{\mu_i}(t)$: Empirical average reward of arm $i$ played upto time t
- UCB-Reward$_{\mathrm{i}}(t)$ = $\widehat{\mu_i}(t) + \sqrt{\left(\frac{8\, logt}{N_i(t)}\right)}$

Each time play the available arm with highest UCB-Reward

# Online Learning: Ripple Effect of Exploration

- **Explore events** decouples Greedy-UCB-Reward and Greedy-Reward

Set of available arms for Greedy-UCB-Reward at time t
$\neq$ Set of available arms for Greedy–Reward at time t




| Arm | $\mu_i$ | $D_i$ |
|---|---|---|
| 1 | 1 | 4 |
| 2 | 1 | 4 |
| 3 | 0.9 | 2 |
| 4 | 0.1 | 1 |

# Online Learning: Action Set Equivalence

- Equality in set of available arms in each time step used in regret analysis of UCB like algorithms

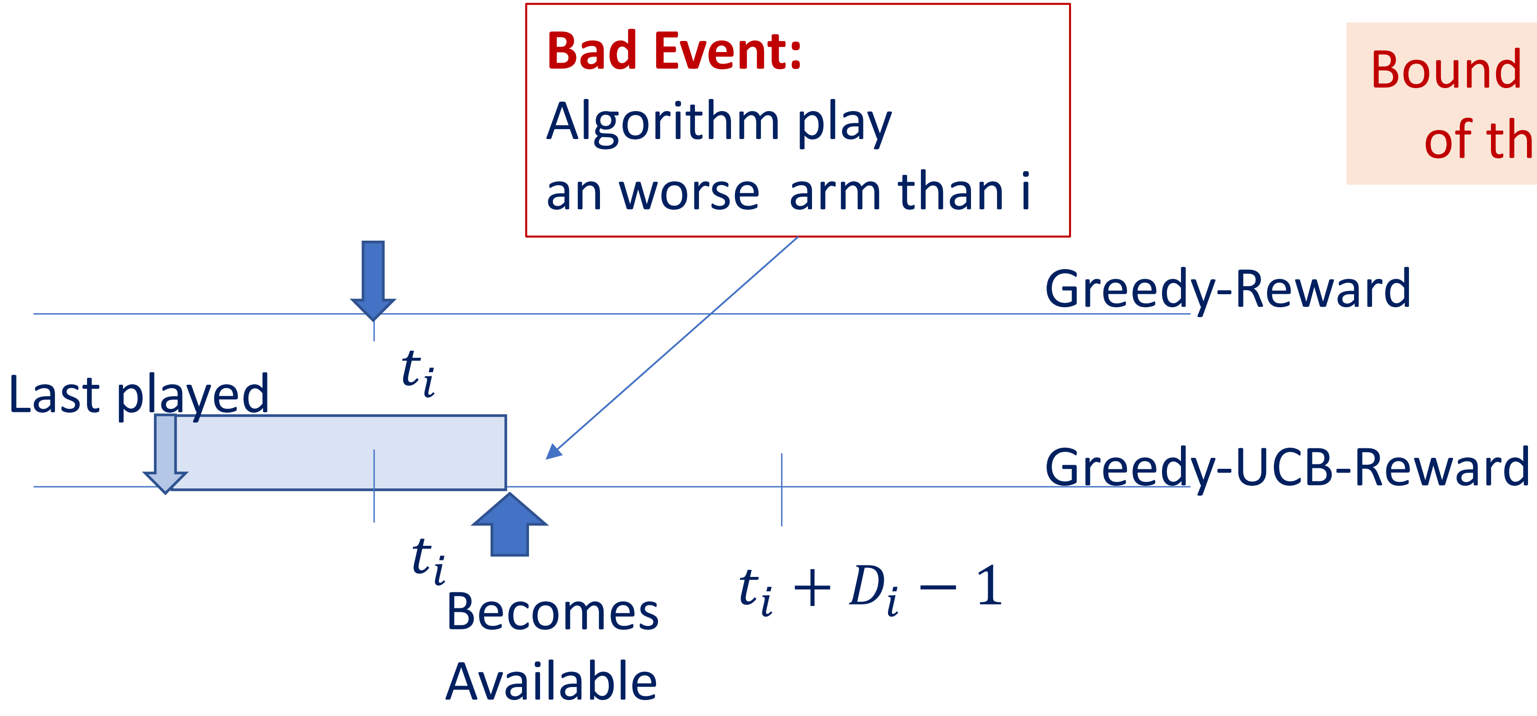
  Multi Armed Bandits: P Auer et al. 02, Sleeping Bandits: R Kleinberg et al. 10 , Combinatorial Bandits: W Chen 13, Combinatorial Semi-Bandits: B Kveton 13

Sleeping Bandits: Arms become busy (go to sleep) but independent of the policy

Sub-optimality in time t only due to estimation error in time t

# Online Learning: Coupling with Greedy

- Strategy in absence of the equality: **Couple Each Arm Separately!**




Bound the occurrences of these bad events

# Online Learning: Free Exploration

- If arm $i$ is available a **worse arm** is played at time t
  - With probability at most $\boldsymbol{O}(\boldsymbol{t}^{-\alpha}), \alpha > 2$, for j $> K^*$ (UCB property)
  - With probability at most $\boldsymbol{O}(\boldsymbol{exp}(-\boldsymbol{ct}))$ for j $\in [i+1, K^*]$ (Free explore)

**UCB Property**: Each arm played $\geq c' \log(t)$ times

$$\widehat{\mu_i}(t) + \sqrt{\left(\frac{2\, logt}{N_i(t)}\right)}$$

**Free explore**: Due to blocking of higher ranked arms, each arm $i \in [1, K^*]$ played $\geq cT$ times up to time T

**Specific to our problem**

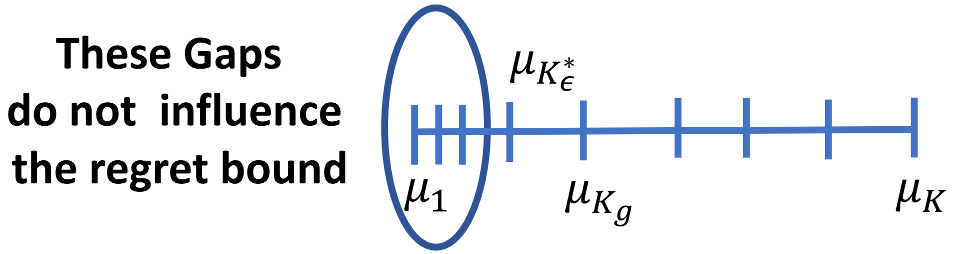
If Arm 1 has delay $D_1 = 4$ then Arm 2 to Arm K is played (in aggregate) at least 75% of time

# Online Learning: Regret Bound

- $K_g$= The highest ranked arm played by Greedy-Reward
- $K_\epsilon^* =$ Lowest ranked arm covering $\left(1-\frac{1}{\epsilon}\right)$ fraction = $\min\left\{j : \sum_{i=1}^{j} \frac{1}{D_i} \geq 1 - \frac{1}{\epsilon}\right\}$

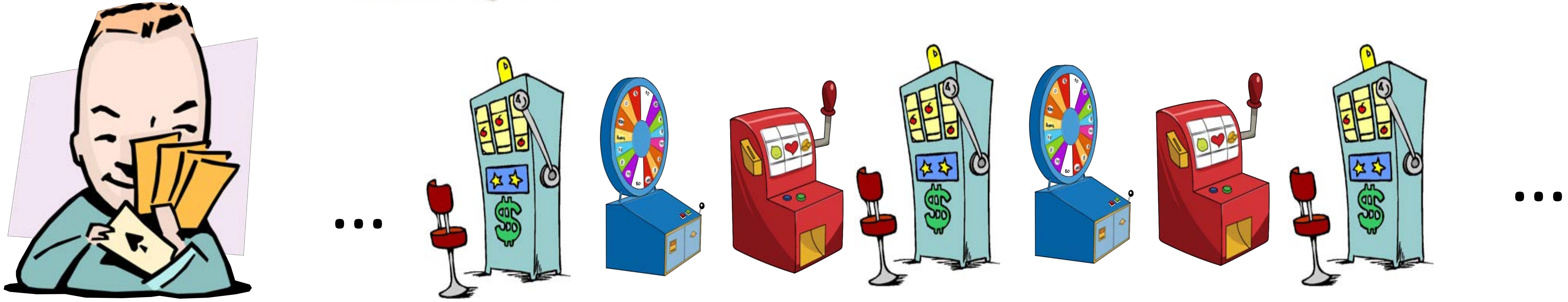
**(1-1/e)-Regret** $= \min_{\epsilon>0} O\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right) + \frac{32K_g(K-K_\epsilon^*)}{\min_{i\in[K_\epsilon^*,\ldots,K_g]}(\mu_i-\mu_{i+1})}\log(T)$

**Result 3**




$$K_g(K-K_\epsilon^*) \leq \min(\mathrm{D_{max}}, \mathrm{K}) \times (\mathrm{K}-(1-\epsilon)\mathrm{D_{min}})$$

# Blocking Bandits


**Free Exploration**

All images/clip arts are taken from open sourced websites

# Future Directions

**Improving Guarantees:**

- Incorporating delays $D_i$ to beat Greedy-Reward (complexity vs gain)
- Improving lower bound using other instances

**Model Extensions**

- *Stochastic Unknown Delay*
- *Multi-type Extension*
  - In each time slot an i.i.d. type is chosen by nature
  - For type j, arm i has delay $D_{ij}$ and reward $\mu_{ij}$
  - Applications: Heterogeneous task allocation, ad placement, recommendation, Ride sharing platform

# Research Overview

- Online Learning: (Design simple and provably near optimal algorithm)
  - Blocking Bandits, Neurips 2019
  - Pareto Optimal Streaming Unsupervised Classification, ICML 2019
  - Switching Constrained Max-weight Scheduling, Infocom 2019
  - Adaptive TTL-based caching for content delivery, Sigmetrics 2017
- Mechanism Design:
  - New Complexity results and Algorithms for Minimum Tollbooth Problem, WINE 2015
  - Reconciling Selfish Routing with Social Good, SAGT 2017
- ML Optimization:
  - Reconciling Adaptive Methods for Over-parameterized Problems*
- Learning Graphical Models:
  - Disentangling Mixture of Epidemics on Graphs*

* Not the primary author

# Thanks

Questions?

# Offline Optimization Problem: Hardness

- **Dense PINWHEEL SCHEDULING (DPWS)** [R. Holte et al. 1989]

- K Arms with Delay $D_i$ for arm $i$ and $\sum_i \frac{1}{D_i}$ = 1 (**dense**)
- **Can we cover 1 to T timeslots by placing the K arms?**
  "Hard" to decide [T. Jacobs and S. Longo 2014]

**YES Instance**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 3 | 1 | 2 | 1 | 3 | 1 |

$\boldsymbol{D_1 = 2, D_2 = 4, D_3 = 4}$

- **Reduction:**

- DPWS instance with Reward = 1 for each arm
- One additional arm with Reward = 0 and Delay = 1

**Is OPT = T? "Hard" to decide** **Result 1**

**"Hard"**:
NO pseudo-polynomial algorithm
Unless SAT is solvable by a
randomized algorithm in expected
$O(n^{\log(n)\log(\log(n))})$ time

# Online Learning: Negative Regret

- Example: Greedy-UCB-Reward performs better than Greedy-Reward

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Greedy-Reward** | 1 | 2 | 3 | | 1 | 2 | 3 | | 1 | 2 | 3 |

| Arm | $\mu_i$ | $D_i$ |
|---|---|---|
| 1 | 1 | 3 |
| 2 | 0.9 | 3 |
| 3 | 0.5 | 2 |

- If the following event occurs (constant probability event)

UCB-Reward_3 > UCB-Reward_1 > UCB-Reward_2

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Greedy-UCB-Reward** | 3 | 1 | 3 | 2 | 3 | 1 | 2 | 3 | 1 | 3 | 2 |

Deadlock
Negative Regret!

# Online Learning: Regret Lower Bound

- **Setting:** $\forall\, i, D_i = D$ and Greedy-Reward is Optimal

$$\lim_{T \to \infty} \text{Regret}/\log(\text{T}) \geq \frac{(K - K_0^*)}{\min_{i \in [K_0^*, \ldots, K_g]} (\mu_i - \mu_{i+1})}$$

- Lower Bound possible only because **Greedy-Reward is optimal**
- Follows from lower bound on learning best-K arms from semi-bandit feedback
  V. Anantharam 1987