# Combinatorial Multi-Armed Bandit with General Reward Functions

Wei Chen Wei Hu Fu Li Jian Li Yu Liu Pinyan Lu

NIPS 2016

# Stochastic Multi-armed Bandit

- A player against $m$ arms
  - select one arm to pull in each round
- Each pulled arm generates a random reward following an unknown distribution
- Observe partial feedbacks
- Goal: collect cumulative reward over multiple rounds as much as possible
- Regret: measure the performance of a bandit algorithm
  - Difference of cumulative reward of optimal solution and the cumulative reward of the bandit strategy

# Combinatorial Multi-armed Bandit

- The player selects a subset of arms (a super arm), collectively provides a random reward to the player
- Semi-bandit feedback
- Applications: Online advertising,  online recommendations, wireless routing
- The action unit is a combinatorial object:
  - A set of advertisements, a route in a wireless network
- The reward depends on unknown stochastic behaviors
  - Users' click through behaviors, wireless transmission quality

# Previous work on CMAB

- Linear reward functions
- Non-linear reward functions
  - The expected reward for playing a super arm is a linear combination/non-linear function of the expected outcomes from the constituent base arms
- Many natural reward functions do not satisfy this property
  - Function $\max()$: its expectation depends on the entire distributions of the input random variables, not just their means
  - $X_1 = X_2 \sim \{0,1\}\, with\ p = 0.5, \mathbb{E}[\max(X_1, X_2)] = 0.75$
  - $Y_1 = Y_2 \sim U(0,1), \mathbb{E}[\max(Y_1, Y_2)] = \frac{2}{3}$

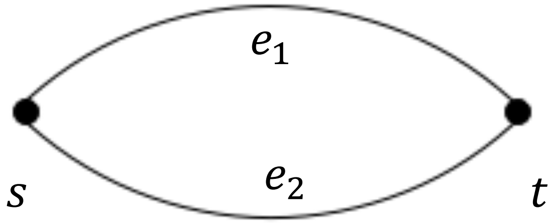
# K-Max Problem

- An application in auctions:
- The auctioneer is repeatedly selling an item to $m$ bidders
- In each round, the auctioneer selects $K$ bidders to bid
- Each of the $K$ bidders independently draws his bid from his private valuation distribution
- The auctioneer uses first-price auction to determine the winner
    - Payment = The largest bid

# Expected Utility Maximization Problem

- Maximizing $\mathbb{E}[u(\sum_{i \in S} X_i)]$
  - $X_i$'s are independent random variables
  - $S$ is decision among all feasible sets
  - $u$ is the utility function
- $X_i$ can be the random delay of edge $e_i$ in a routing graph
- $S$ is a routing path in the graph
- $u$ is non-linear to model risk-averse/risk-prone behaviors
  - No longer a function of the means of underlying random variables

# Rationale Behind EUM



- A graph with two nodes $s$ and $t$, two parallel links $e_1$ and $e_2$
- $e_1$ has a fixed length 1
- $e_2$ has a length of 0.9 with probability 0.9 and a length of 1.9 with probability 0.1
- Risk-averse user: choose $e_1$, $u(x) = \begin{cases} 1, x \leq 1 \\ 0, x > 1 \end{cases}$
- Risk-prone user: choose $e_2$, $u(x) = \frac{1}{x+1}$

# Problem formulation

- A set of base arms: $E = [m]$
- A set of subsets: $\mathcal{F} \subseteq 2^E$
- A probability distribution $D$ over $[0,1]^m$
- Stochastic outcomes: $X = (X_1, \ldots, X_m) \sim D$
- A reward function: $R$: $[0,1]^m \times \mathcal{F} \to \mathbb{R}^+$
    - Only depends on the revealed outcomes
- A super arm: each feasible subset of arms $\mathrm{S} \in \mathcal{F}$
- Expected reward of choosing a super arm $S$: $r_D(S) = \mathbb{E}_{X \sim D}[R(X, S)]$

# Benchmark

- When the distribution $D$ is known, the optimal algorithm is to choose the optimal super arm in each round
    - $S^* = \mathrm{argmax}_{S\in\mathcal{F}} r_D(S)$
- May be computationally hard to find the optimal super arm
- $\alpha$ −approximation regret
- $Reg(T) = T \cdot \alpha \cdot r_D(S^*) - \sum_{t=1}^{T} r_D(S_t)$

# Assumptions

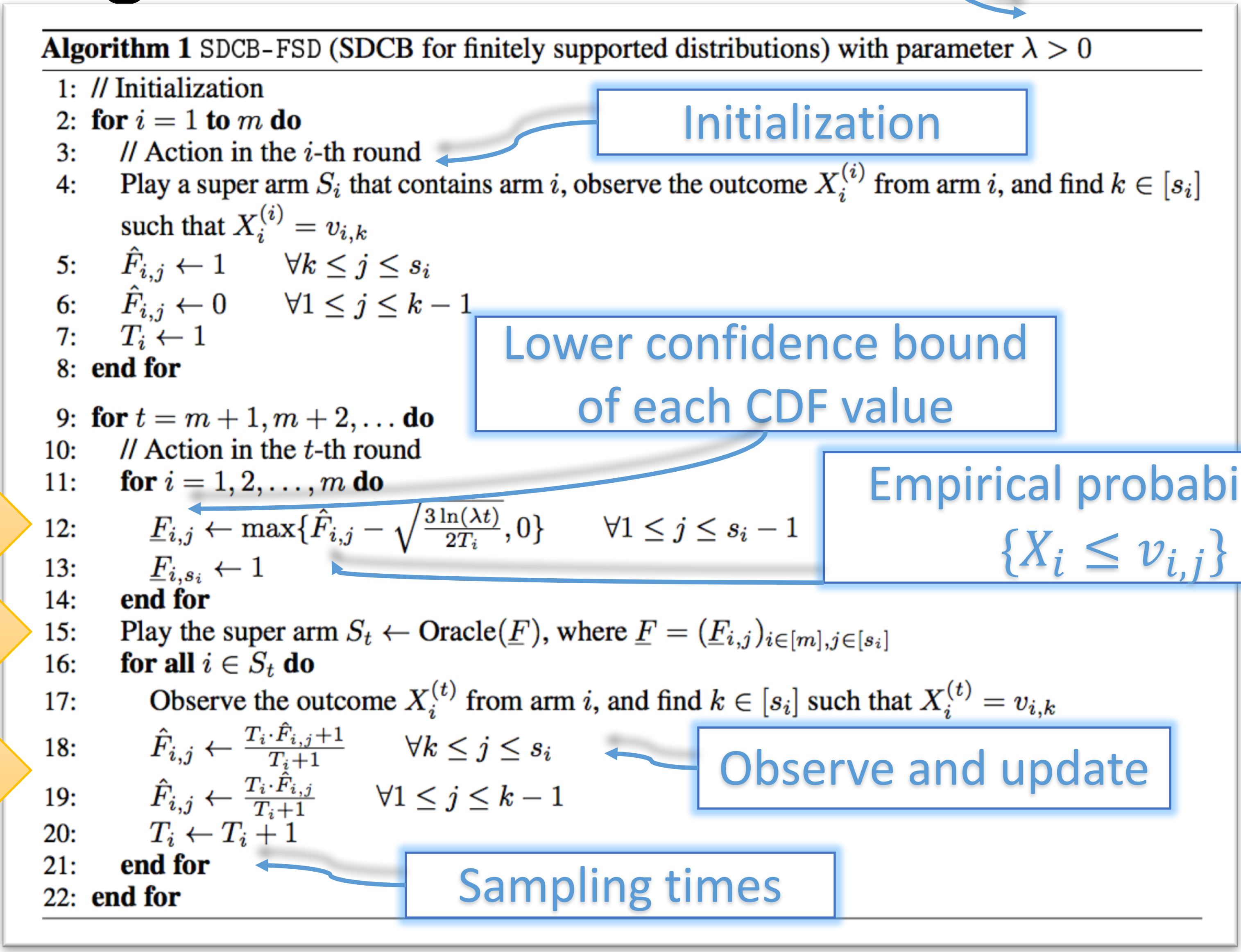
- Independent outcomes from arms
- Bounded reward value
- Monotone reward function
  - $R(x, S) \leq R(x', S)\ if\ x_i \leq x_i'$
- Lipschitz-continuous reward function
  - $|R(x, S) - R(x', S)| \leq C\ \sum_{i \in S} |x_i - x_i'|$
- Require an $\alpha$ −approximation computation oracle to produce decisions

# Discrete Distributions

- Known finite support
  - $supp(D_i) = \{v_{i,1}, v_{i,2}, \dots, v_{i,s_i}\}, \forall i \in [m]$
- $D_i$ can be fully described by its CDF values
  - $F_{i,j}^{D} = \Pr_{X_i \sim D_i}[X_i \leq v_{i,j}], \forall j \in [s_i]$
- The computation oracle takes a CDF vector as an input and output an approximated solution

# Algorithm SDCB

Control the confidence radius



**Algorithm 1** SDCB-FSD (SDCB for finitely supported distributions) with parameter $\lambda > 0$

```
// Initialization
for $i = 1$ to $m$ do
    // Action in the $i$-th round
    Play a super arm $S_i$ that contains arm $i$, observe the outcome $X_i^{(i)}$ from arm $i$, and find $k \in [s_i]$
    such that $X_i^{(i)} = v_{i,k}$
    $\hat{F}_{i,j} \leftarrow 1 \quad \forall k \le j \le s_i$
    $\hat{F}_{i,j} \leftarrow 0 \quad \forall 1 \le j \le k-1$
    $T_i \leftarrow 1$
end for

for $t = m+1, m+2, \ldots$ do
    // Action in the $t$-th round
    for $i = 1, 2, \ldots, m$ do
        $\underline{F}_{i,j} \leftarrow \max\{\hat{F}_{i,j} - \sqrt{\frac{3\ln(\lambda t)}{2T_i}}, 0\} \quad \forall 1 \le j \le s_i - 1$
        $\underline{F}_{i,s_i} \leftarrow 1$
    end for
    Play the super arm $S_t \leftarrow \text{Oracle}(\underline{F})$, where $\underline{F} = (\underline{F}_{i,j})_{i \in [m], j \in [s_i]}$
    for all $i \in S_t$ do
        Observe the outcome $X_i^{(t)}$ from arm $i$, and find $k \in [s_i]$ such that $X_i^{(t)} = v_{i,k}$
        $\hat{F}_{i,j} \leftarrow \frac{T_i \cdot \hat{F}_{i,j} + 1}{T_i + 1} \quad \forall k \le j \le s_i$
        $\hat{F}_{i,j} \leftarrow \frac{T_i \cdot \hat{F}_{i,j}}{T_i + 1} \quad \forall 1 \le j \le k-1$
        $T_i \leftarrow T_i + 1$
    end for
end for
```

Initialization

Lower confidence bound of each CDF value

Empirical probability of $\{X_i \le v_{i,j}\}$

S1

S2

S3

Observe and update

Sampling times

# Algorithm SDCB

- Idea: Optimism in the face of uncertainty principle
- A smaller $F_{i,j}$ means the larger realization has a higher probability
- With high probability each $\underline{D}_i$ has first-order stochastic dominance over $D_i$
  - The distribution $F$ first-order stochastically dominates $G$ iff $F(x) \leq G(x), \forall x$
  - $F$ gives at least as high a probability of receiving at least $x$ as does $G$
- Monotonicity $\Rightarrow r_{\underline{D}}(S) \geq r_D(S), \forall S$ with high probability
- $\underline{D}$ provides an optimistic estimation on the expected reward of each super arm

# Proof Sketch

- Regret bound: $O(\log T)$ distribution-dependent
- Three terms in regret:
- Initialization stage
- When an inaccurate estimation happens
    - The number of bad rounds can be bounded by Chernoff bound
    - $\sum_t \frac{1}{t^2}$
- All base arms are accurately estimated
    - Sampling threshold

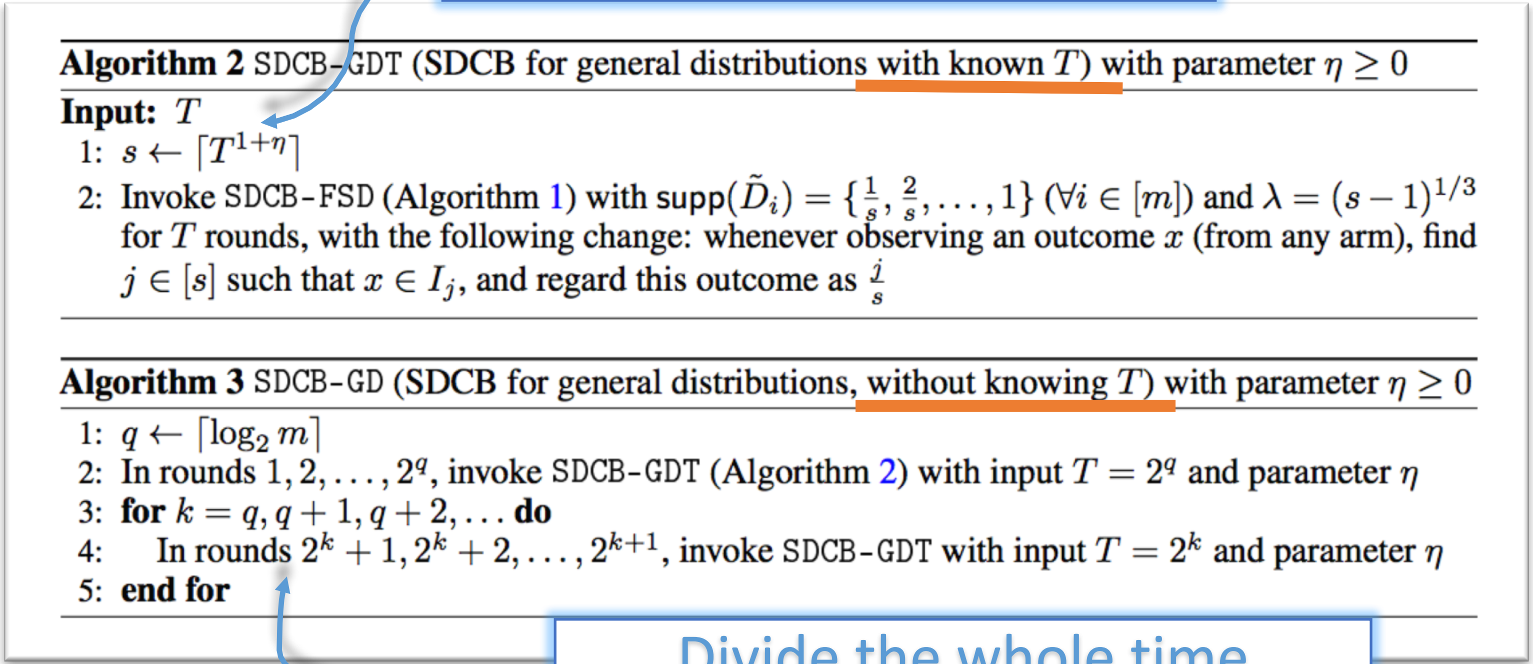
# Compare with CMAB

- The mean value of $\underline{D}_i$ is close to the expectation with high probability
  - By Chernoff bound
- The previous analysis can be applied to SDCB
  - Nearly the same regret bound

# General Distributions

- A discretization step on distributions → Apply SDCB algorithm

Discretization parameter



**Algorithm 2** SDCB-GDT (SDCB for general distributions with known $T$) with parameter $\eta \geq 0$

```
Input: $T$
$s \leftarrow \lceil T^{1+\eta} \rceil$
Invoke SDCB-FSD (Algorithm 1) with $\mathrm{supp}(\tilde{D}_i) = \{\frac{1}{s}, \frac{2}{s}, \ldots, 1\}$ $(\forall i \in [m])$ and $\lambda = (s-1)^{1/3}$
    for $T$ rounds, with the following change: whenever observing an outcome $x$ (from any arm), find
    $j \in [s]$ such that $x \in I_j$, and regard this outcome as $\frac{j}{s}$
```

**Algorithm 3** SDCB-GD (SDCB for general distributions, without knowing $T$) with parameter $\eta \geq 0$

```
$q \leftarrow \lceil \log_2 m \rceil$
In rounds $1, 2, \ldots, 2^q$, invoke SDCB-GDT (Algorithm 2) with input $T = 2^q$ and parameter $\eta$
for $k = q, q+1, q+2, \ldots$ do
    In rounds $2^k + 1, 2^k + 2, \ldots, 2^{k+1}$, invoke SDCB-GDT with input $T = 2^k$ and parameter $\eta$
end for
```

Divide the whole time horizon into periods

# General Distributions

- When the time horizon $T$ is known in advance
- Perform a discretization on $D$ to get a discrete distribution $\widetilde{D}$
- Partition $[0,1]$ evenly into $s$ intervals: $I_1, \dots, I_s$
    - $\Pr_{\tilde{X}_i \sim \widetilde{D}_i}[\tilde{X}_i = \frac{j}{s}] = \Pr_{X_i \sim D_i}[X_j \in I_j]$
- Pretend that the outcomes are drawn from $\widetilde{D}$ instead of $D$
    - Replacing any outcome $x \in I_j$ by $\frac{j}{s}$
- The discretization parameter $s$ depends on $T$

# General Distributions

- When the time horizon $T$ is unknown in advance
- Use doubling trick to avoid the dependency on $T$
  - Partition time horizon into periods of exponentially increasing lengths and run the original algorithm on each period
  - Whenever we reach a round $t$ such that $t$ is a power of 2, restart the algorithm, forgetting all of the information gained in the past
  - At the expense of a constant factor

# Proof Sketch

- $O(\log T)$ distribution-dependent regret and $O(\sqrt{T \log T})$ distribution-independent regret
- The regret consists of two parts
- The regret for the discretized CMAB problem
- The error due to discretization
    - Lipschitz continuous property
    - The lengths of discretized intervals

# Summary

- A new problem: learning the shape of the distribution
- Previous work has strong assumptions
  - Bernoulli distribution, single parametric distribution with prior information
- Use UCB on CDF instead of mean value
- Could be a comparison to the current work