The Ohio State University

AT&T

UCBoost: A Boosting Approach to Tame Complexity and Optimality for Stochastic Bandits

Fang Liu[1], Sinong Wang[1], Swapna Buccapatnam[2] and Ness Shroff[1]

[1]The Ohio State University

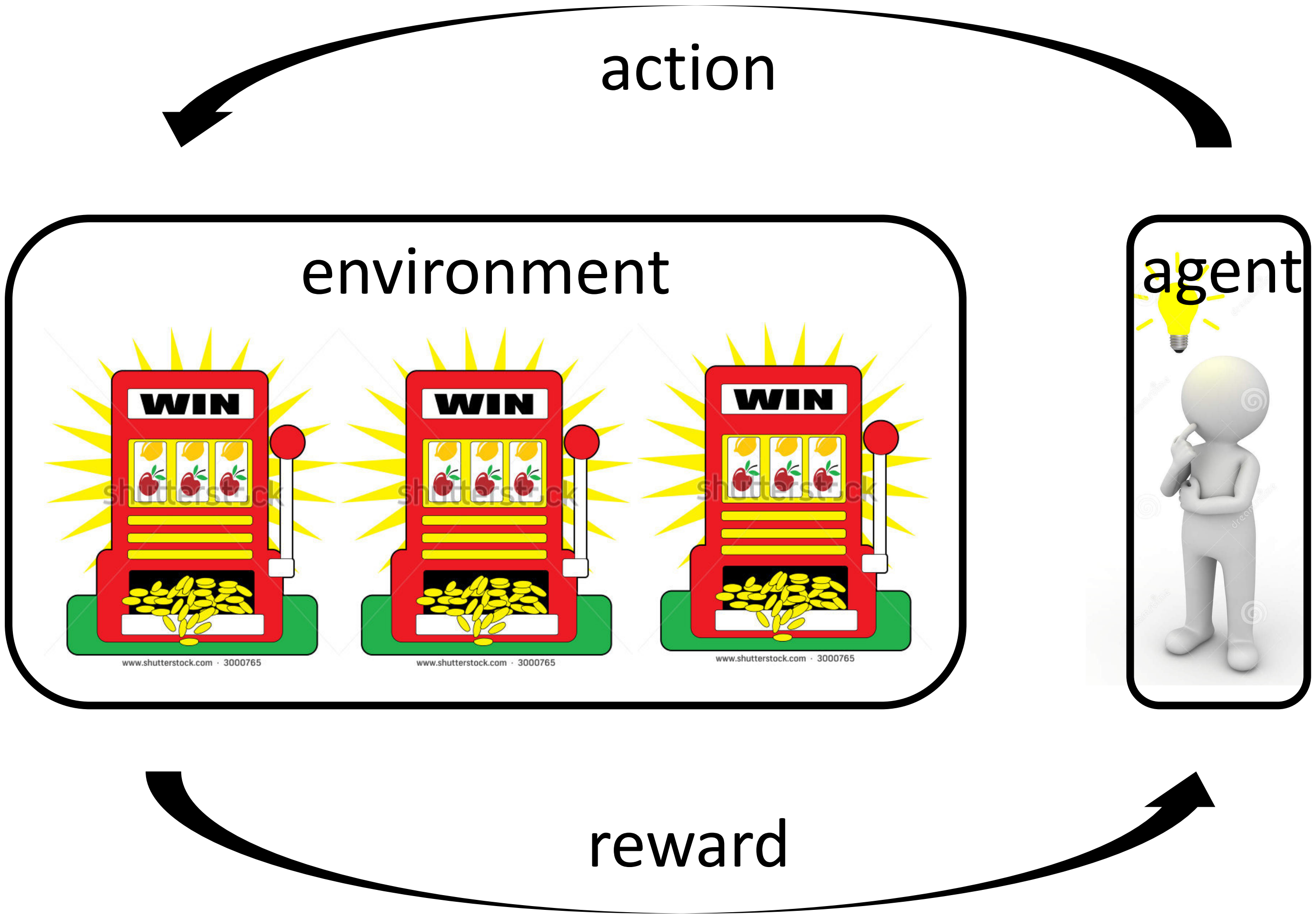
[2]AT&T Labs Research

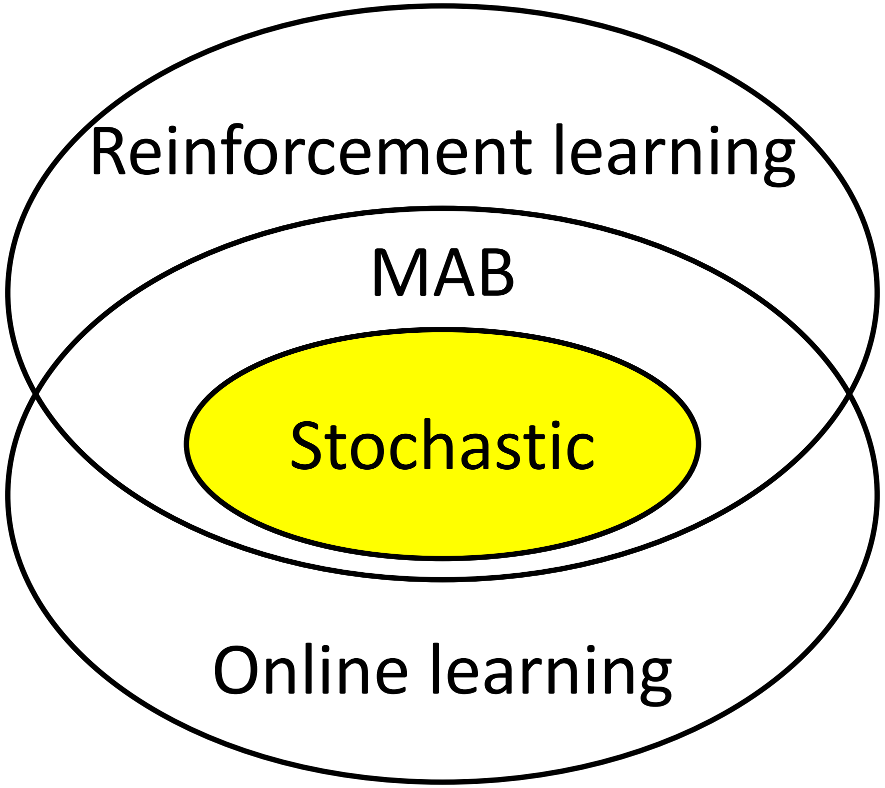
Outline

- Background and Motivations
  - ❑Multi-Armed Bandits Framework
  - ❑Stochastic Bandits At a Glance
  - ❑Complexity vs Optimality Dilemma
  - ❑Our results Overview
- UCBoost Algorithm
  - ❑Generic UCB Algorithm
  - ❑UCBoost(D) Algorithm
  - ❑UCBoost(ϵ) Algorithm
- Numerical Results
  - ❑Experiment Setting
  - ❑Regret Results
  - ❑Computation Results
- Conclusion

Multi-Armed Bandits Framework

- Repeated game between an agent and an environment





Stochastic Bandits At a Glance

- Model
  - At each (discrete) time t, the agent plays action A_t from a set of K actions
  - The agent receives reward $Y_{A_t,t}$, drawn from unknown distribution A_t
- Performance measure
  - Regret(loss) $$R(T) = \mathbb{E}\left[\max_{i\in[K]} \sum_{t=1}^{T} Y_{i,t} - \sum_{t=1}^{T} Y_{A_t,t}\right]$$
  - Minimize regret = maximize total reward
- Regret lower bounds
  - Problem-dependent: $\Omega\left(\sum_i \frac{\mu^* - \mu_i}{KL(\mu_a, \mu^*)} \log T\right)$ where μ_i is expected reward
- Popular algorithms
  - Upper Confidence Bounds (UCB), Thompson Sampling, epsilon-greedy

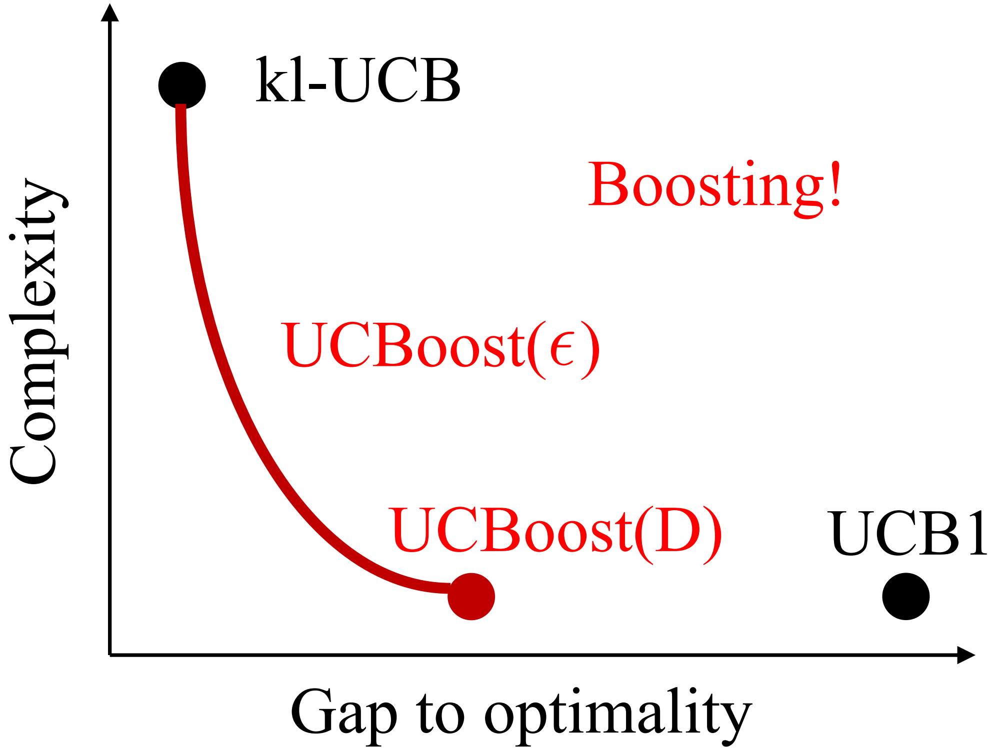
Complexity vs Optimality Dilemma

- Computational Complexity matters
  - Real-time applications: robotic control, portfolio optimization
  - Large-scale applications: recommendation systems, meta-algorithm for learning
- Optimal algorithms involve heavy computation ☹
  - kl-UCB, DMED: optimization problems
  - Thompson Sampling: posterior updating
- Simple algorithms are far from being optimal ☹
  - UCB1: gap of Pinsker’s inequality is unbounded
  - Epsilon-greedy: same gap as UCB1, requires one more prior knowledge

Can we design an algorithm that can trade-off complexity and optimality?

Our Results Overview

- UCBoost algorithms
  - Ensemble a set of “weak” but closed-form UCB-type algorithms
  - Propose two solutions: a finite set and an infinite set for any epsilon
  - First to offer trade-off between complexity and optimality with guarantees



	kl-UCB	UCBoost(ϵ)	UCBoost(D)	UCB1
Regret/log(T)	$O\left(\sum_a \frac{\mu^* - \mu_a}{d_{kl}(\mu_a, \mu^*)}\right)$	$O\left(\sum_a \frac{\mu^* - \mu_a}{d_{kl}(\mu_a, \mu^*) - \epsilon}\right)$	$O\left(\sum_a \frac{\mu^* - \mu_a}{d_{kl}(\mu_a, \mu^*) - 1/e}\right)$	$O\left(\sum_a \frac{\mu^* - \mu_a}{2(\mu^* - \mu_a)^2}\right)$
Complexity	unbounded	$O(\log(1/\epsilon))$	$O(1)$	O(1)

Outline

- Background and Motivations
  - ❑Multi-Armed Bandits Framework
  - ❑Stochastic Bandits At a Glance
  - ❑Complexity vs Optimality Dilemma
  - ❑Our results Overview
- UCBoost Algorithm
  - ❑Generic UCB Algorithm
  - ❑UCBoost(D) Algorithm
  - ❑UCBoost(ϵ) Algorithm
- Numerical Results
  - ❑Experiment Setting
  - ❑Regret Results
  - ❑Computation Results
- Conclusion

Generic UCB Algorithm

- Semi-distance function $d : \Theta \times \Theta \to \mathbb{R}$
  - Between expectations of random variables over bounded support Θ
  - Non-negative, triangle inequality, not necessary to be symmetric
  - Strong semi-distance function satisfies $d(p, q) = 0 \text{ iff } p = q$
- kl-dominated: upper-bounded by $d_{kl}(p, q) = p \log \frac{p}{q} + (1 - p) \log \frac{1 - p}{1 - q}$
- Generic UCB algorithm is

At time t, play arm $\arg\max_{a \in \mathcal{K}} \max\{q \in \Theta : N_a(t) d(\bar{Y}_a(t), q) \leq \log(t)\}$

Theorem 1. If d is a strong semi-distance function and is also kl-dominated, then the regret of UCB(d) algorithm is

$$\limsup_{T \to \infty} \frac{\mathbb{E}[R(T)]}{\log T} \leq \sum_a \frac{\mu^* - \mu_a}{d(\mu_a, \mu^*)}$$

Generic UCB Algorithm

- UCB kernel is a semi-distance function d, with problem
  $$P(d) : \max_{q \in \Theta} \quad q \qquad s.t. \quad d(p,q) \leq \delta$$
  - kl-UCB: kl-divergence d_{kl}, need iterative method to solve
  - UCB1: $d_{sq}(p,q) = 2(p-q)^2$, closed-form solution
- New semi-distance functions with closed-form solutions
  - Hellinger distance: $d_h(p,q) = (\sqrt{p} - \sqrt{q})^2 + \left(\sqrt{1-p} - \sqrt{1-q}\right)^2$
  - Biquadratic distance: $d_{bq}(p,q) = 2\,(p-q)^2 + \frac{4}{9}\,(p-q)^4$
  - Theorem 1 provides regret bounds for these new UCB algorithms
  - Closed-form solution allows O(1) complexity
- A natural question is

Can we ensemble these closed-form UCB algorithms to a "stronger" one?

UCBoost(D) Algorithm

- Consider a set D of kl-dominated semi-distance functions. If $\max_{d \in D} d$ is a strong semi-distance function, then D is said to be feasible
  - Sufficient condition: exists one strong semi-distance in D
  - Easy to construct and verify a feasible set ☺
- UCBoost(D) algorithm is

At time t, play arm $\arg\max_{a \in \mathcal{K}} \min_{d \in D} \max\{q \in \Theta : N_a(t) d(\bar{Y}_a(t), q) \leq \log(t)\}$

Theorem 2. If D is a feasible set of kl-dominated semi-distance functions, then the regret of UCBoost(D) algorithm is

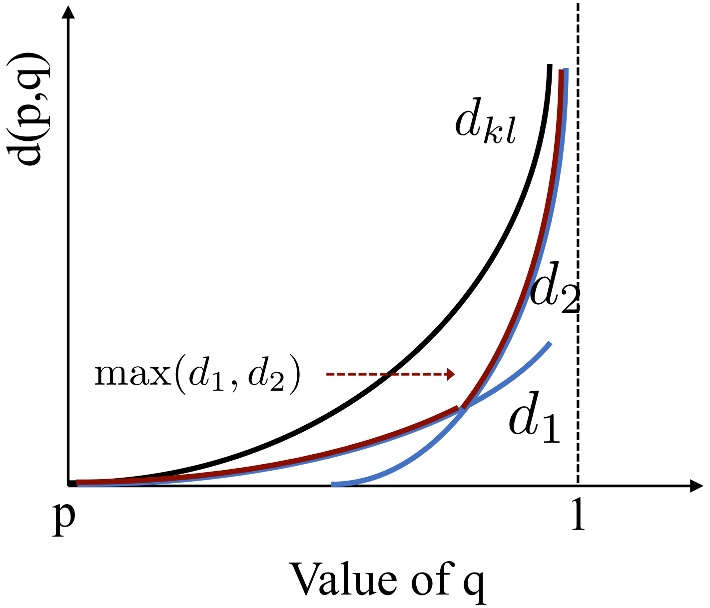
$$\limsup_{T \to \infty} \frac{\mathbb{E}[R(T)]}{\log T} \leq \sum_a \frac{\mu^* - \mu_a}{\max_{d \in D} d(\mu_a, \mu^*)}$$

- If all d in D have closed-form solutions, complexity is O(|D|)

UCBoost(D) Algorithm

- Why taking the minimum?
- Philosophy of voting
  - Majority vote?
  - No! (If the ordering is known, follow the leader)
  - UCBoost takes the minimum, thus the tightest upper confidence bound
- Geometric view of UCBoost
  - Kernel of UCBoost is $\max_{d \in D} d$
  - Taking the minimum = solving $P\left(\max_{d \in D} d\right)$
  - The closer to KL divergence, the better the regret

	UCB1	UCB2	UCB3	UCBoost
	0.9	0.8	0.6	0.6
	0.8	0.75	0.7	0.7
	0.2	0.2	0.3	0.2
decision	1	1	2	2



UCBoost(D) Algorithm

- A new candidate semi-distance function
  - Lower bound of d_{kl}: $d_{lb}(p,q) = p\log(p) + (1-p)\log\frac{1-p}{1-q}$
  - Closed-form solution of $P(d_{lb})$
  - Tight to d_{kl} when q goes to 1
  - Allows bounded gap to optimality

Corollary 1. If $D=\{d_{bq},d_h,d_{lb}\}$, then the regret of UCBoost(D) algorithm is

$$\limsup_{T\to\infty}\frac{\mathbb{E}[R(T)]}{\log T} \leq \sum_a \frac{\mu^* - \mu_a}{d_{kl}(\mu_a,\mu^*) - 1/e}$$

where e is the natural number. The complexity is O(1) per arm per round.

UCBoost(ϵ) Algorithm

- Recall geometric view of UCBoost:
  - The closer to KL divergence, the better the regret
  - Design a sequence of semi-distance functions to approximate d_{kl}
- For any ϵ, step-function approximation
  - A sequence of points: $q_k = 1 - 1/(1+\epsilon)^k$ for any $k \geq 0$
  - For each k, step function: $d_s^k(p,q) = d_{kl}(p,q_k)1\{q > q_k\}$
  - For each p, construct dynamic set $D(p) = \{d_{sq}, d_{lb}, d_s^k : p \leq q_k \leq \exp(-\epsilon/p)\}$
  - Bisection search over step functions in D(p)

Theorem 3. The regret of UCBoost(ϵ) algorithm is

$$\limsup_{T\to\infty} \frac{\mathbb{E}[R(T)]}{\log T} \leq \sum_a \frac{\mu^* - \mu_a}{d_{kl}(\mu_a, \mu^*) - \epsilon}$$

The complexity is $O(\log(1/\epsilon))$ per arm per round.

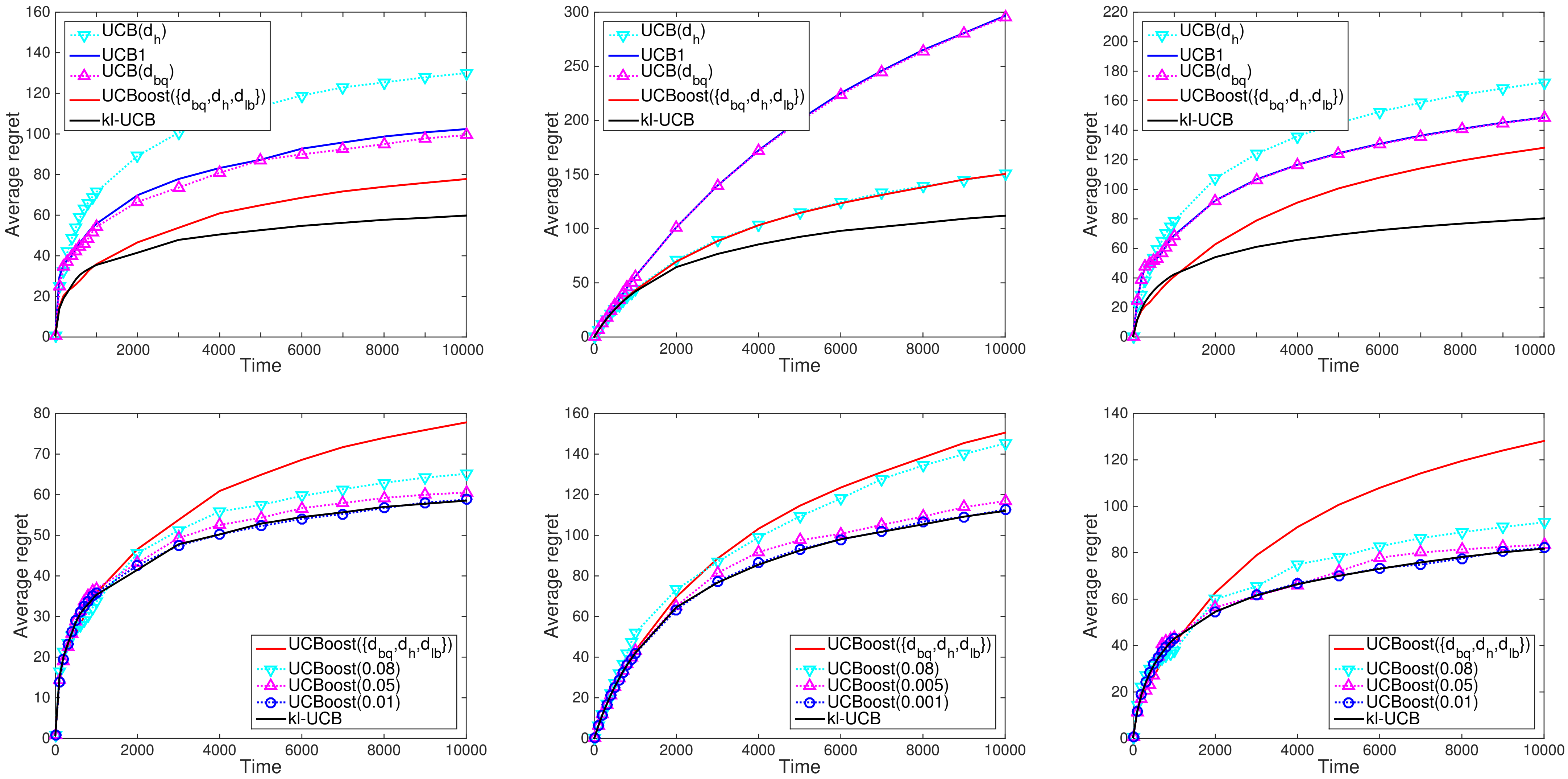
Outline

- Background and Motivations
  - ❑Multi-Armed Bandits Framework
  - ❑Stochastic Bandits At a Glance
  - ❑Complexity vs Optimality Dilemma
  - ❑Our results Overview
- UCBoost Algorithm
  - ❑Generic UCB Algorithm
  - ❑UCBoost(D) Algorithm
  - ❑UCBoost(ϵ) Algorithm
- Numerical Results
  - ❑Experiment Setting
  - ❑Regret Results
  - ❑Computation Results
- Conclusion

Experiment Setting

- Average results over 10k independent runs of the algorithms
- Bernoulli Scenario 1
  - 9 arms with μ_i = i/10
  - Basic scenario with Bernoulli rewards
- Bernoulli Scenario 2
  - 10 arms with $\mu_1 = \mu_2 = \mu_3 = 0.01, \mu_4 = \mu_5 = \mu_6 = 0.02, \mu_7 = \mu_8 = \mu_9 = 0.05, \mu_{10} = 0.1$
  - Model the cases in online recommendations
- Beta Scenario
  - 9 arms with Beta distributions, Beta(a_i,2), where a_i = i
  - Another typical distribution with bounded support

Regret Results



(a) Bernoulli scenario 1 (b) Bernoulli scenario 2 (c) Beta scenario

Computation Results

- Computational Costs per arm per round

Scenario	kl-UCB	UCBoost(ϵ) $\epsilon = 0.01(0.001)$	UCBoost(ϵ) $\epsilon = 0.05(0.005)$	UCBoost(ϵ) $\epsilon = 0.08$	UCBoost($\{d_{bq}, d_h, d_{lb}\}$)	UCB1
Bernoulli 1	$933\mu s$	$7.67\mu s$	$6.67\mu s$	$5.78\mu s$	$1.67\mu s$	$0.31\mu s$
Bernoulli 2	$986\mu s$	$8.76\mu s$	$7.96\mu s$	$6.27\mu s$	$1.60\mu s$	$0.30\mu s$
Beta	$907\mu s$	$8.33\mu s$	$6.89\mu s$	$5.89\mu s$	$2.01\mu s$	$0.33\mu s$

  - UCBoost(D) always outperforms UCB1 with same scale of computational cost
  - 1% computation cost of kl-UCB to achieve competitive regret
  - 100x faster response time or 100x capacity of arms

Conclusion

- Generic UCB algorithm
  - New alternatives to UCB1
- Two recipes for complexity vs optimality dilemma
  - UCBoost(D) algorithm: bounded gap, O(1) complexity
  - UCBoost(ϵ) algorithm: ϵ-gap, O(log(1/ϵ) complexity
- A boosting framework
  - Design of UCB algorithm reduces to finding new semi-distance functions
  - Try your own semi-distance functions

End

Thanks!