The Irrevocable Multi-Armed Bandit Problem

Ritesh Madan
Qualcomm-Flarion Technologies

May 27, 2009

Joint work with Vivek Farias (MIT)

Multi-Armed Bandit Problem

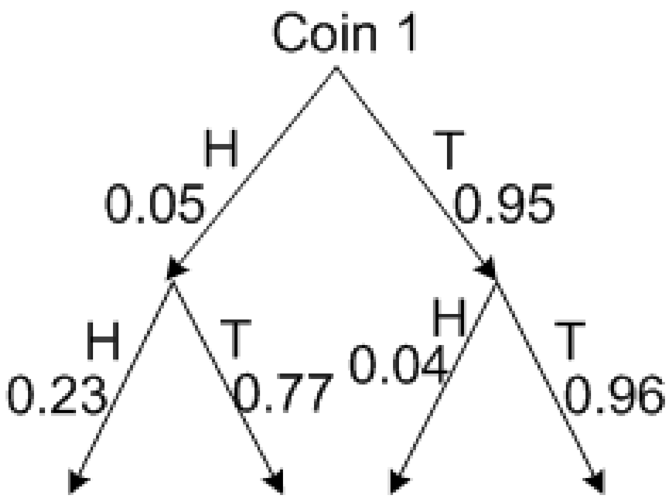
- n arms, where each arm i is a Markov Decision Process (MDP)
  - state space $\mathcal{S}_i$
  - action space $\mathcal{A}_i$
  - reward function $r_i(s_i, a_i)$
  - transition probability from s_i to s_i' under action a_i is $P(s_i, a_i, s_i')$
  - idle action ϕ_i with zero reward, unchanged state
- *Constraint:* k arms can be pulled at each time step.
- *Goal:* Maximize expected reward over a finite horizon, T
- *Applications:* call center staffing, fast fashion retailing, clinical drug trials

Example: Flipping Coins With Uncertain Bias

- n coins, each with uncertain bias $p_i \in [0, 1]$, where p_i is Pr(Heads)
- Can flip up to k coins at each time
  - action space $\mathcal{A}_i = \{\text{flip}, \phi\}$
- For every flip of coin i
  - $\$1$ if heads, 0 if tails
  - refine estimate of p_i
- When coin is not flipped, no reward and no refinement of estimate of bias
- *Goal:* Compute policy for flipping to maximize expected reward over T time steps.

Exploitation vs Exploration

- Tradeoff between exploiting a reliable coin and exploring another coin with potentially high reward.
- Assume a conjugate prior for a two-coin example below (e.g., Bernoulli-Beta learning model)



Whittle's Heuristic

- *Subsidy for idling:* Set $r_i(s_i, \phi_i) = \lambda$, for all s_i

- At time t, if arm is in state $s_i(t)$, compute minimum value of λ for this arm such that the optimal action in state $s_i(t)$ is to idle
  - call this value $\eta_i(s_i(t))$

- At time t, pull k arms with the highest $\eta_i(s_i(t))$'s computed above

- Good performance on average, but lots of "churn"
  - Example sample path for 5 binomial coins, 10 time steps, 2 pulls at each time shown below

t	0	1	2	3	4	5	6	7	8	9
coin 1	1	3	5	1	4	1	3	5	5	5
coin 2	2	4	2	3	5	2	4	2	3	4

Irrevocability: Fast Fashion Retailing

- *Fast Fashion Retailing:* Adjust assortment offered on sale at the store to quickly adapt to popular fashion trends
- Issues with Whittle's heuristic
  - each new run introduces fixed cost
  - if product is likely to come back, disincentive to buy now
- *Constraint:* Once a product is off the shelf, it won't come back, i.e., can pull an arm *only* if either
  - the arm was pulled in the last time step, *or*
  - the arm was never pulled in the past
- *Questions:*
  - is irrevocability a tractable constraint?
  - what is the price of irrevocability?

Key Results

- Packing heuristic for multi-armed bandit problem
  - k arms pulled simultaneously
  - reward earned by a single bandit depends on number of pulls, i.e., *value is correlated with size*

- A uniform bound on price of irrevocability for an interesting (large) class of bandits

- Computational experiments show that irrevocability can lead to loss of less than 10 to 20 percent in practice

- Construct a fast computational algorithm to compute packing heuristic
  - faster than Whittle's heuristic

Prior Work: Stochastic Knapsack, Dean et al. [06]

- n items with values $v_1, \ldots, v_n$ and unknown (random) sizes $s_1, \ldots, s_n$ with known means
- Consider the following LP

$$\text{max.} \quad \left\{ \sum_i x_i v_i : \sum_i x_i \mathbb{E}[s_i] \leq t, x_i \in [0,1] \right\}$$

  - A solution is to set $x_i = 1$ for bandits with highest $v_i/\mathbb{E}[s_i]$
  - Greedy approximation algorithms based on placing items in (essentially) the following order:

$$\frac{v_1}{\mathbb{E}[s_1]} \geq \ldots \geq \frac{v_n}{\mathbb{E}[s_n]}$$

- Analysis relies critically on the fact that *the value is independent of the size*

Prior Work: Budgeted Learning, Guha and Munagala [07]

- n coins with uncertain reward
  - *Exploration:* k arms can be played sequentially
  - *Exploitation:* one arm is selected to be played forever
  - design exploration strategy to maximize reward during exploitation
- Treat each bandit as an item in the knapsack
  - value is expected reward if exploited
  - two size constraints: cost, exploitation
  - expected reward of arm is independent of length of exploration
- Policy based on LP where size constraints are met in expectation

Related Work: Index Based Policies, Goel et al. [08]

- Index based policy for budgeted learning that is within constant factor of optimal
  - faster computation compared to Guha and Munagala
  - index is constant factor approximation of Gittin's index (and vice versa) for appropriate discount factor
  - Gittin's index obtains constant factor approximation for budgeted learning
- Extensions to finite horizon multi-armed bandit problem

LP Relaxation for Multi-Armed Bandit Problem

- Relax the problem by removing irrevocability constraint, and over time horizon T, allow
$$\mathbb{E}(\text{total pulls}) = kT$$

- Problem becomes tractable LP

$$\begin{aligned} \text{maximize} \quad & \sum_i (\text{expected reward for } i \text{ under } \pi_i) \\ \text{subject to} \quad & \sum_i (\text{expected pulls for } i \text{ under } \pi_i) \leq kT \\ & \pi_i \in D_i \end{aligned}$$

  where π_i is state-action frequency for arm i, constrained to be in a polytope of permissible state-action frequencies, D_i.

- Fast computation via dual later...

Packing Heuristic

- Each arm is an item of value $\mathbb{E}[R_i]$ and size $\mathbb{E}[T_i]$
  - R_i is the (random) reward earned by arm i under policy π_i^*
  - T_i is the (random) number of pulls for arm i under π_i^*
- Order arms as

$$\frac{\mathbb{E}[R_1]}{\mathbb{E}[T_1]} \geq \frac{\mathbb{E}[R_2]}{\mathbb{E}[T_2]} \geq \ldots \geq \frac{\mathbb{E}[R_n]}{\mathbb{E}[T_n]}$$

- Start with top k arms
- At each time t, pull or idle according to policy for given arm
  - if arm is pulled, increment its local time, t_i, by one
  - if arm is idled, increment time t_i for that arm until another pull action is found or $t_i = T$
  - discard arm once $t_i = T$, replace with next highest ranked arm

Uniform Bound

- Correlation between pulls and reward satisfies *decreasing returns property*

$$\mathbb{E}[R_i^{m+1}] - \mathbb{E}[R_i^m] \leq \mathbb{E}[R_i^m] - \mathbb{E}[R_i^{m-1}]$$

where R_i^m is the reward earned by first m pulls of arm i under optimal policy π_i^* for arm i, for the relaxed LP.

- Above property satisfied by learning problems

- For bandits with decreasing returns property,

$$J^{\mu_{\text{packing}}} \geq \frac{1}{8} J^*$$

where J^* is optimal value of objective function of relaxed LP.

Proof Outline

- Define

$$h = \min\left\{ j : \sum_{i=1}^{j} E[T_i] \geq kT/2 \right\} \wedge \min\left\{ i : \sum_{j=1}^{i} T_j \geq kT/2 \right\}$$

- Show (using techniques similar to Dean et al., Guha & Munagala)

$$\mathbb{E}\left[\sum_{i=1}^{h} R_i\right] \geq \frac{1}{4} OPT(RLP(\tilde{\pi}_0))$$

- The first h bandit obtains expected reward of at least $\mathbb{E}\left[\sum_{i=1}^{h} R_i\right]/2$
  - decreasing rewards property
  - a simple combinatorial lemma to show that each bandit $\leq h$ is pulled for at least $T/2$ steps

Numerical Computation: Model

- Each bandit is modeled as a coin with unknown bias
  - Bernoulli arrivals
- The prior for the coin is assumed to be a Beta distribution parameterized by (α, β)
  - conjugate prior for Bernoulli arrivals
  - mean number of arrivals per time slot is $\alpha/(\alpha + \beta)$
- *Update:*

$$\alpha_i = \alpha_i + \mathbf{1}_{[\text{arrival}]}, \qquad \beta_i = \beta_i + \mathbf{1}_{[\text{no arrival}]}$$

- Coefficient of variation (CV) represents uncertainty in coin bias:

$$cv = \frac{\sigma}{\mu}$$

Performance

Horizon (T)	Arms (n)	Pulls (k)	Performance: J^{μ}/J^{*}			Revocations
			Packing	Whittle Irrev	Whittle	Whittle
40	501	125	0.91	0.80	0.92	1983
40	99	25	0.91	0.80	0.92	389
40	501	75	0.88	0.80	0.91	1055
40	99	15	0.88	0.79	0.90	214

Equal number of bandits with cvs 1, 2.5, 4.

Fast Computation

- Solving relaxed LP via interior point methods is roughly $O(nTA\Sigma)^3$
  - Σ states, A actions per arm

- We derive a computational algorithm with complexity $O(nA\Sigma^2 \log(kT))$ per time step
  - compare with $O(TnA\Sigma^2 \log(kT))$ per time step for index based Whittle's heuristic

- Policy is essentially a randomization between two index policies
  - indices computed only at start; no updates at each time step necessary

Dual Problem

- Consider the LP relaxation

$$\begin{aligned} \text{maximize} \quad & \sum_i R_i(\pi_i) \\ \text{subject to} \quad & \sum_i T_i(\pi_i) \leq kT \\ & \pi_i \in D_i \end{aligned}$$

- Dual problem given by

$$\begin{aligned} \text{minimize} \quad & \lambda kT + \sum_i \max_{\pi_i \in D_i} (R(\pi_i) - \lambda T_i(\pi_i)), \\ \text{subject to} \quad & \lambda \geq 0 \end{aligned}$$

Dual Decomposition

Dual program is

$$\begin{aligned} \text{minimize} \qquad & \lambda k T + \sum_i \max_{\pi_i \in D_i} (R(\pi_i) - \lambda T_i(\pi_i)), \\ \text{subject to} \qquad & \lambda \geq 0 \end{aligned}$$

- Bisection algorithm to compute λ^*
  - $\log(kT)$ iterations; at iteration k solve, for each arm i,

$$\max_{\pi_i \in D_i} (R_i(\pi_i) - \lambda_k T_i(\pi_i))$$

  - dynamic programming can be used for above computation, complexity of $O(A\Sigma^2 T)$ for A actions, Σ states
  - need bisection to converge to λ such that corresponding state-action frequencies satisfy $\sum_i T_i(\pi_i) \approx kT$

Non Differentiable Dual

- Consider two bandits, $T = 1$, one pull.

$$\begin{array}{ll} \text{maximize} & R(p) = p_1 + p_2 \\ \text{subject to} & T(p) = p_1 + p_2 \le 1 \end{array}$$

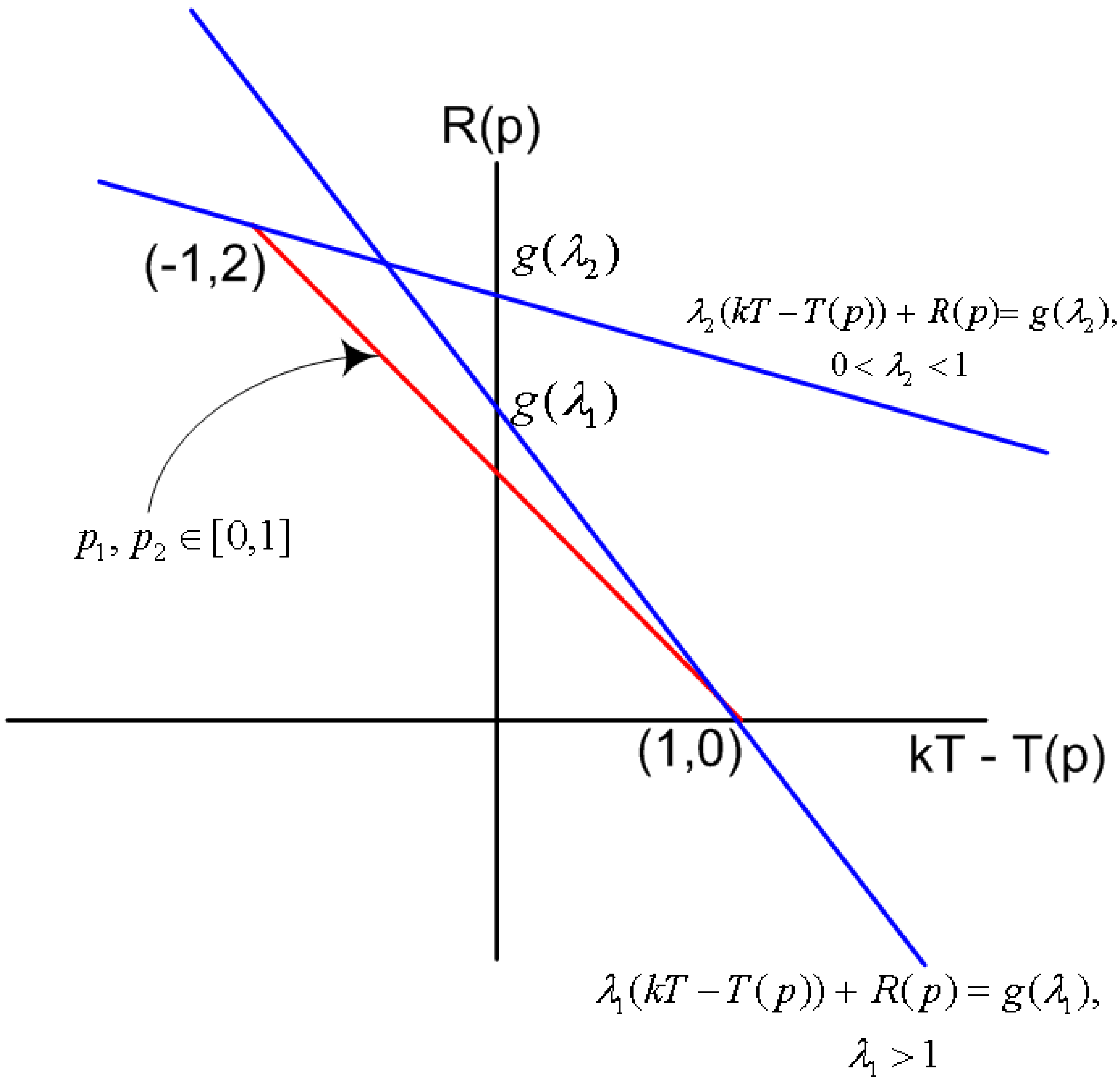
- Dual function is

$$g(\lambda) = \max_{p_1, p_2} (R(p) + \lambda(T(p) - 1))$$

$$= \begin{cases} 2 - \lambda \ , & \lambda \le 1 \\ \lambda \ , & \lambda > 1 \end{cases}$$

- For $\lambda > 1$, budget exceeded by one pull; for $\lambda < 1$, zero pulls.

Primal Solution via Dual



An Optimal Policy: Linear Combination of Policies

- Consider

$$\lambda_1 \in (\lambda^*, \lambda^* + \epsilon] \qquad \text{and} \qquad \lambda_2 \in [\lambda^* - \epsilon, \lambda^*]$$

- $\pi(\lambda) = \text{arg max}_{\pi_i \in D_i}(R_i(\pi_i) - \lambda T_i(\pi_i))$

- Consider a linear solution of corresponding optimal state action frequencies:

$$\pi = \alpha\pi(\lambda_1) + (1 - \alpha)\pi(\lambda_2)$$

  where $\alpha \in [0, 1]$ is chosen such that

$$kT = \alpha T(\lambda_1) + (1 - \alpha)T(\lambda_2)$$

- π is feasible, and the reward earned is guaranteed to be within 2ϵ of optimal.

Summary

- Designed an irrevocable packing heuristic which performs well in practice

- For bandits with decreasing rewards,

  - uniform constant factor $(1/8)$ approximation

  - upper bound on price of irrevocability

- Derived a fast computational scheme to compute the packing heuristic