An Optimal Index Policy for the Multi-Armed Bandit Problem with Re-Initializing Bandits

Peter Jacko*

YEQT III
November 20, 2009

*Basque Center for Applied Mathematics (BCAM), Bilbao, Spain
Example: Congestion Control in Router
Multi-Armed Bandit Problem
Multi-Armed Bandit Problem

- A classic problem of **efficient learning**
- Originally in sequential design of experiments
  - Thompson (1933): which of two drugs is superior?
  - Robbins (1952), Bradt et. al (1956), Bellman (1956)
- Job sequencing problem
  - Cox & Smith (1961): $c\mu$-rule
- Celebrated general solution
  - Gittins and colleagues (1970s): Gittins index rule
  - crucial condition: non-played are frozen
Outline

• Re-initializing bandits
  ▶ Transmission Control Protocol (TCP)

• MDP formulation

• Relaxations and decomposition into subproblems

• Optimal solution to the subproblems
  ▶ obtaining an index

• Optimal solution to relaxations

• Optimal index policy to original problem
Transmission Control Protocol (TCP)

- Implemented at the two ends of a connection
- A way of end-to-end congestion control
- Provides reliable, ordered delivery of a stream of packets
- Fully-sensitive to packet losses
- Examples: web browsers, e-mail, file transfer (FTP)
- Must be distinguished from congestion control in routers
- An alternative (UDP) is used for VoIP, streaming, etc.
TCP End-to-End Connection

- Sender sends an initial packet and waits
- Receiver sends acknowledgment of each received packet
- Sender sends more packet(s) after receiving acknowledgment(s) or restarts after time-out
TCP Dynamics as Markov Chain

• States: \( n \in \{0, 1, \ldots, N - 1\} = \text{sending rate level} \)
  - \( n = 0 \): sending rate of 1 packet/RTT
  - \( n = N - 1 \): maximum rate, \( \leq W_{\max} \)

• Transitions: OK (acknowledgment), NO (time-out)
Congestion Control of TCP Flows in Router

- Time epochs $t = 0, 1, 2, \ldots$

- Two possible control actions $a(t)$:
  - transmit the flow packets
  - block the flow by dropping packets

- If transmitted $W_{X(t)}$ packets in state $X(t)$, then
  - goodput (reward) $R_{X(t)}$ is earned
  - the sender sets $X(t + 1)$ given by TCP dynamics

- **Objective:** Maximize the long-run goodput (reward)
  - while choosing exactly one flow every time epoch
Congestion Control in Router

- Maximizing time-average expected goodput

\[
\max_{\pi \in \Pi} \lim_{T \to \infty} \frac{1}{T} \mathbb{E}_{\pi}^{\pi} \left[ \sum_{t=0}^{T-1} \sum_{m \in \mathcal{M}} R_{m, X_m(t)}^{a_m(t)} \right]
\]

- Subject to sample path condition

\[
\sum_{m \in \mathcal{M}} a_m(t) = 1, \text{ for all } t
\]

conditional on state history under \( \pi \)
Relaxations

1: Whittle’s Relaxation: choose one on average

\[
\lim_{T \to \infty} \frac{1}{T} \mathbb{E}^\pi_n \left[ \sum_{t=0}^{T-1} \sum_{m \in \mathcal{M}} a_m(t) \right] = 1
\]

2: Multiply by \( W^{\max} \) and use \( W^{am(t)}_{m,Xm(t)} \leq W^{\max} a_m(t) \),

\[
\lim_{T \to \infty} \frac{1}{T} \mathbb{E}^\pi_n \left[ \sum_{t=0}^{T-1} \sum_{m \in \mathcal{M}} W^{am(t)}_{m,Xm(t)} \right] \leq W^{\max}
\]

3: Dualize this constraint using Lagrangian multiplier

\[
\max_{\pi \in \Pi} \lim_{T \to \infty} \frac{1}{T} \mathbb{E}^\pi_n \left[ \sum_{t=0}^{T-1} \sum_{m \in \mathcal{M}} \left( R^{am(t)}_{m,Xm(t)} - \nu W^{am(t)}_{m,Xm(t)} \right) \right] + \nu W^{\max}
\]
Decomposition

- Decompose the Lagrangian relaxation due to flow independence into single-flow parametric subproblems

\[
\max_{\pi_m \in \Pi_m} \lim_{T \to \infty} \frac{1}{T} \mathbb{E}_{\pi_m}^{\pi_m} \left[ \sum_{t=0}^{T-1} \left( R^{a_m(t)}_{m,X_m(t)} - \nu W^{a_m(t)}_{m,X_m(t)} \right) \right]
\]

- This is known as a restless bandit

- Under certain natural conditions, there exist break-even values \( \nu_{m,n} \) of \( \nu \), called transmission indices (prices), s.t.
  - it is optimal to transmit if \( \nu_{m,n} \geq \nu \)
  - it is optimal to block if \( \nu_{m,n} \leq \nu \)
Optimal Solutions to Relaxations

- For 3 (multi-flow Lagrangian relaxation): For each flow,
  - it is optimal to transmit if $\nu_{m,n} \geq \nu$
  - it is optimal to block if $\nu_{m,n} \leq \nu$

- Suppose that re-initializing state 0 has highest value, i.e., $\nu_{m,0} \geq \nu_{m,n}$ for all states $n$ of any flow $m$

- Denote by $\nu^*$ the second-highest $\nu_{m,0}$ over $m$

- For 1: an optimal policy is: at every $t$,
  - transmit each flow satisfying $\nu_{m,X(t)} > \nu^*$
  - if no such flow exists, then transmit one flow satisfying $\nu_{m,X(t)} = \nu^*$
Optimal Solution to Original Problem

• An optimal policy is: at every $t$,
  ▶ transmit each flow satisfying $\nu_{m,n} > \nu^*$
  ▶ if no such flow exists, then transmit one flow satisfying $\nu_{m,X(t)} = \nu^*$

• This policy chooses exactly one flow every time epoch

• It is optimal here because it is feasible here and optimal for a relaxation

• (See animation)
Multi-Armed Bandit Problem

• We can apply the same reasoning to the classic problem
  ▶ Set the threshold to the second-highest Gittins index
  ▶ Play the bandits with Gittins index higher than the threshold, breaking ties choosing one arbitrarily
  ▶ Once no bandits are above, restart the procedure

• Optimal policy = sequence of optimal solutions to Lagrangian relaxations with decreasing values of Lagrangian multiplier
Routing: A More Realistic Setting

- Bandwidth $W$, i.e., deterministic “server capacity”
- Target time-average router throughput $\overline{W} < W$, i.e., “virtual capacity”
- Buffer size $B \geq W$
- Backlog process $B(t)$ at epochs $t$
  ▶ number of packets buffered for more than one period
- To be allocated to randomly appearing and disappearing flows
Summary

• Apart from multi-armed bandit problem and its special cases, proving optimality of an index policy is rare

• The approach leads to a new proof for the classic problem

• For more complex (restless) bandit problems
  ▶ gives some intuition for when an index policy is optimal
  ▶ presents a well-grounded method for design of (suboptimal) greedy rules
  ▶ useful for problems with on-average constraint (if capacity can be marketed between periods)
Thank you for your attention!
Congestion Control in Router

\[ \begin{align*}
X_1(t) &\xrightarrow{W_{\text{sent}1,X_1(t)}} a_1(t) \\
X_2(t) &\xrightarrow{W_{\text{sent}2,X_2(t)}} a_2(t) \\
X_3(t) &\xrightarrow{W_{\text{sent}3,X_3(t)}} a_3(t) \\
&\quad \vdots \\
X_M(t) &\xrightarrow{W_{\text{sent}M,X_M(t)}} a_M(t) \\
\quad &\xrightarrow{W^{a_1(t)\text{,}_1,X_1(t)}} B(t) \\
&\quad \xrightarrow{W^{a_2(t)\text{,}_2,X_2(t)}} B(t) \\
&\quad \xrightarrow{W^{a_3(t)\text{,}_3,X_3(t)}} B(t) \\
&\quad \vdots \\
&\quad \xrightarrow{W^{a_M(t)\text{,}_M,X_M(t)}} B(t) \\
\quad &\xrightarrow{R^{a_1(t)\text{,}_1,X_1(t)}} \quad \\
&\quad \xrightarrow{R^{a_2(t)\text{,}_2,X_2(t)}} \quad \\
&\quad \xrightarrow{R^{a_3(t)\text{,}_3,X_3(t)}} \quad \\
&\quad \vdots \\
&\quad \xrightarrow{R^{a_M(t)\text{,}_M,X_M(t)}} \quad \\
\end{align*} \]