Understanding Memory-Regret Trade-off for Streaming Stochastic Multi-Armed Bandits

Yuchen He, Zichun Ye and Chihao Zhang

Shanghai Jiao Tong University

2025.1

The Multi-Armed Bandit

The Multi-Armed Bandit



https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The Multi-Armed Bandit

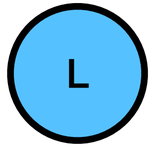
Two Arms

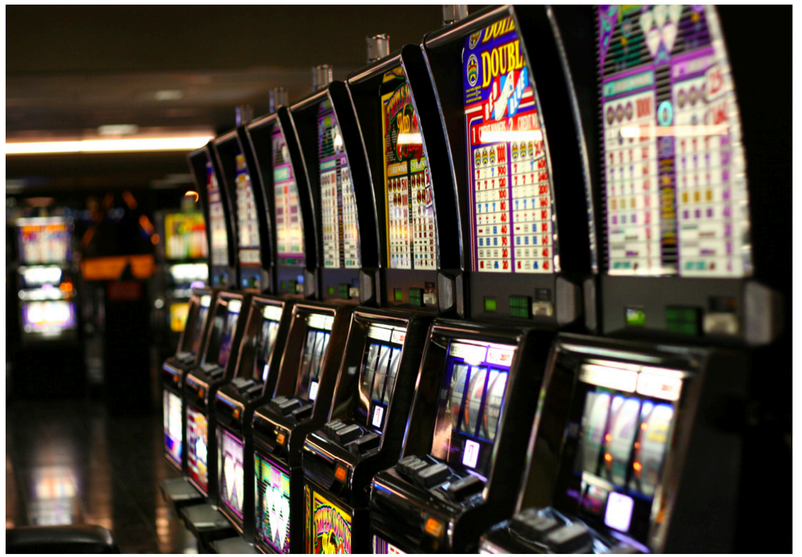


https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The Multi-Armed Bandit

Two Arms

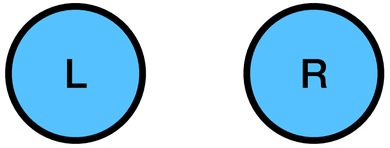




https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The Multi-Armed Bandit

Two Arms

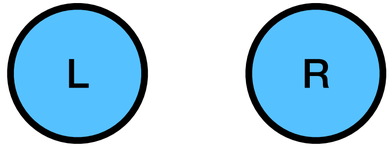




https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The Multi-Armed Bandit

Two Arms



$r_t(L) \sim \texttt{Ber}(0.5)$

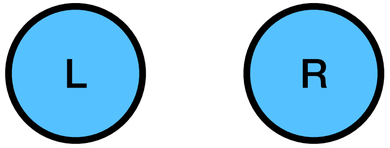
$r_t(R) \sim \texttt{Ber}(0.51)$



https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The Multi-Armed Bandit

Two Arms



$$r_t(L) \sim \texttt{Ber}(0.5)$$

$$r_t(R) \sim \texttt{Ber}(0.51)$$

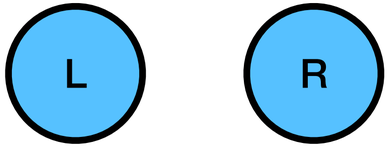
Round	1	2	3	4	5	6	7	8	9
Left	$1	$0		$0					
Right			$1		$1	$0	$0		



https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The Multi-Armed Bandit

Two Arms



$$r_t(L) \sim \texttt{Ber}(0.5)$$

$$r_t(R) \sim \texttt{Ber}(0.51)$$

Round	1	2	3	4	5	6	7	8	9
Left	$1	$0		$0					
Right			$1		$1	$0	$0		

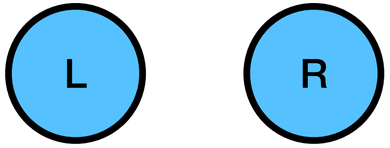


https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The regret

The Multi-Armed Bandit

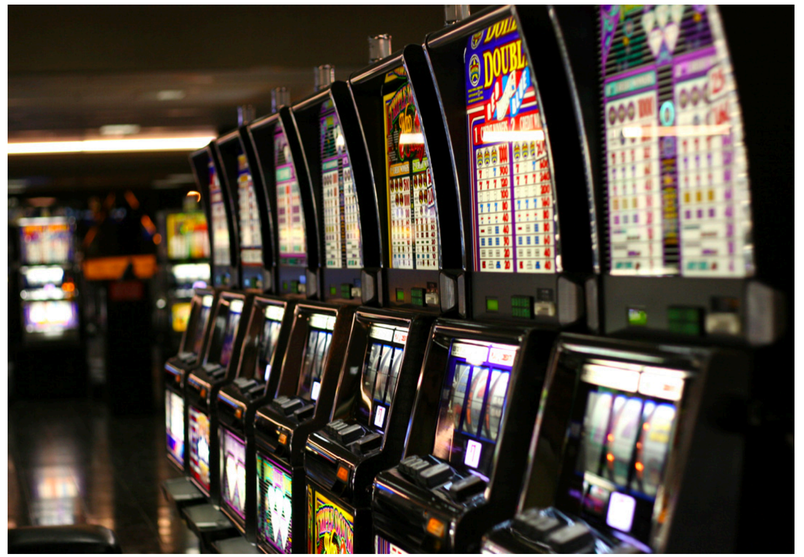
Two Arms



$$r_t(L) \sim \texttt{Ber}(0.5)$$

$$r_t(R) \sim \texttt{Ber}(0.51)$$

Round	1	2	3	4	5	6	7	8	9
Left	$1	$0		$0					
Right			$1		$1	$0	$0		



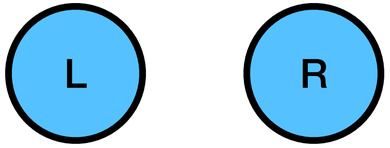
https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

The regret

$$R(T) = \mathbf{E}\left[\sum_{t=1}^{T} r_t(A^*) - r_t(A_t)\right]$$

The Multi-Armed Bandit

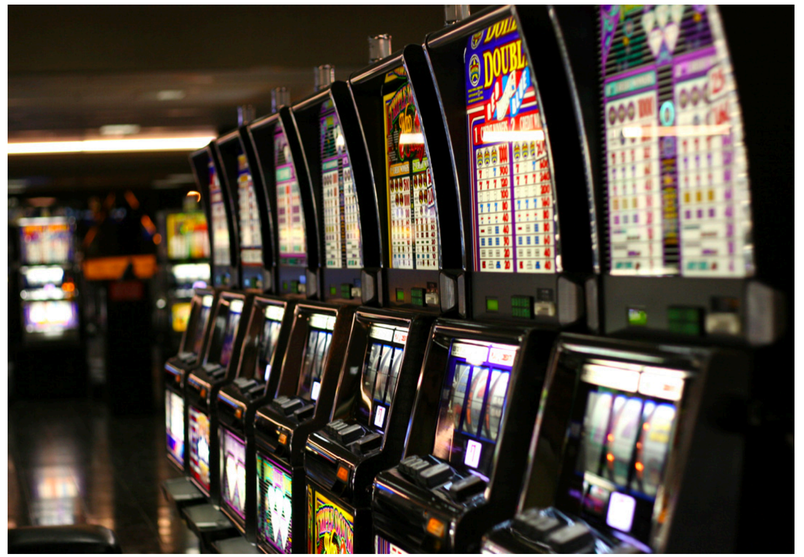
Two Arms



$$r_t(L) \sim \texttt{Ber}(0.5)$$

$$r_t(R) \sim \texttt{Ber}(0.51)$$

Round	1	2	3	4	5	6	7	8	9
Left	$1	$0		$0					
Right			$1		$1	$0	$0		



https://commons.wikimedia.org/wiki/File:Las_Vegas_slot_machines.jpg

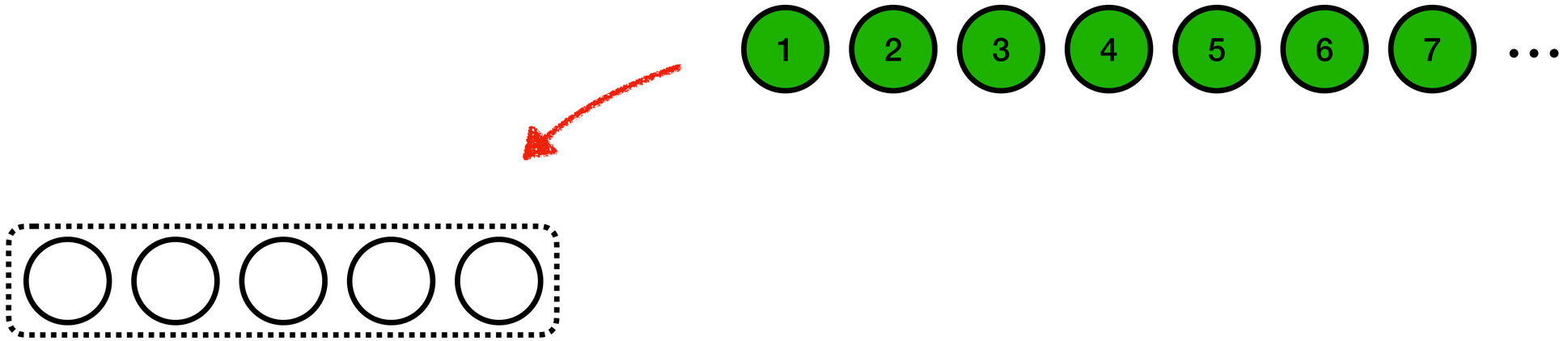
The regret

$$R(T) = \mathbf{E}\left[\sum_{t=1}^{T} r_t(A^*) - r_t(A_t)\right]$$

How to choose $A_t \in \{L, R\}$?

Streaming Model

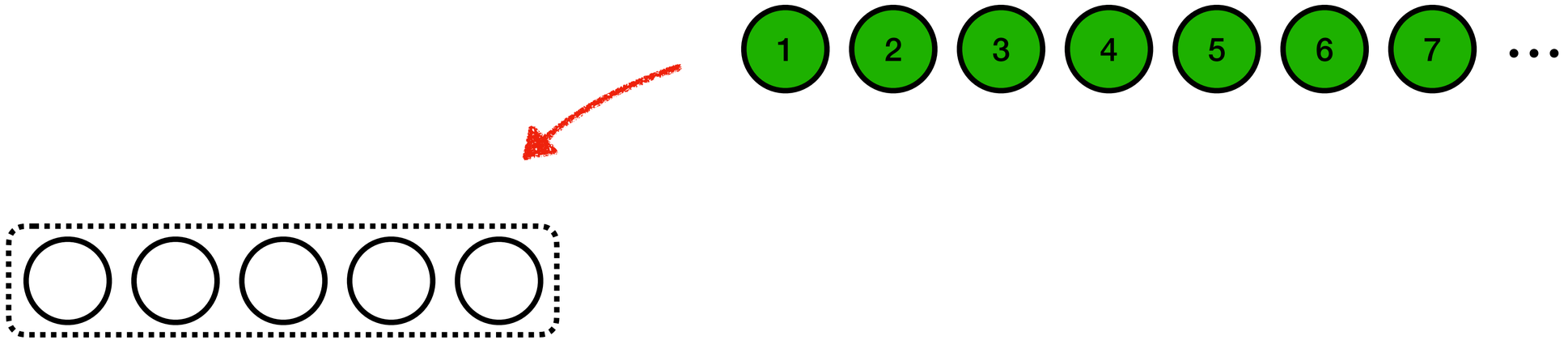
Bounded memory size:



Streaming Model

Bounded memory size:

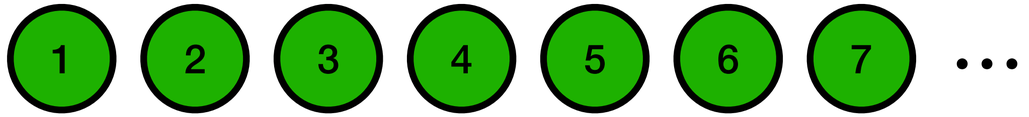
n arms come in a stream:



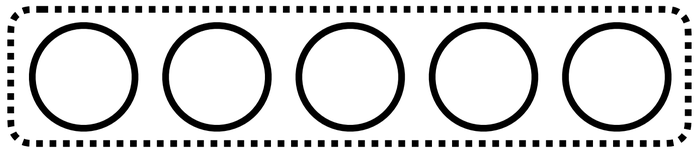
Streaming Model

Bounded memory size:

n arms come in a stream:



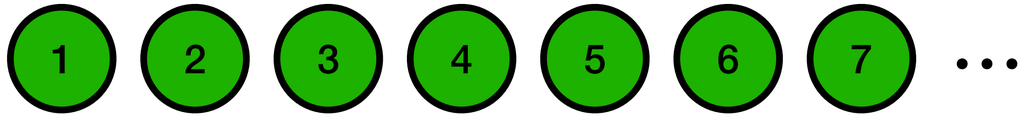
store at most m arms



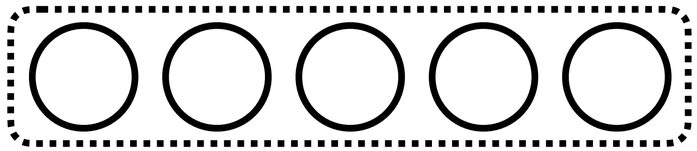
Streaming Model

Bounded memory size:

n arms come in a stream:



store at most m arms



Only arms in the memory can be pulled…

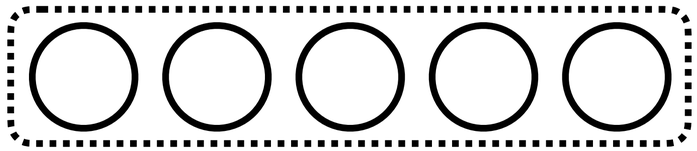
Streaming Model

Bounded memory size:

n arms come in a stream:



store at most m arms

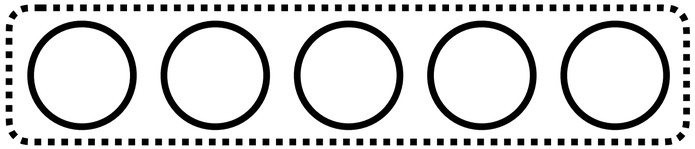


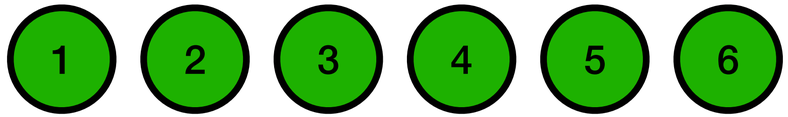
Only arms in the memory can be pulled…

The trade-off between Regret / Memory / # of Passes

A One Pass Example…

6 arms with memory size 5





A One Pass Example…

6 arms with memory size 5



A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

Explore for L rounds and determine which arm to drop

A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

Explore for L rounds and determine which arm to drop

A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

Explore for L rounds and determine which arm to drop

Read in the last arm

A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

Explore for L rounds and determine which arm to drop

Read in the last arm

A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

Explore for L rounds and determine which arm to drop

Read in the last arm

Exploit for $T - L$ rounds

A One Pass Example…

6 arms with memory size 5



Read in the first $n - 1$ arms

Explore for L rounds and determine which arm to drop

Read in the last arm

Exploit for $T - L$ rounds

Classic MAB Algorithm

A One Pass Example…

6 arms with memory size 5



Read in the first $n-1$ arms

Explore for L rounds and determine which arm to drop

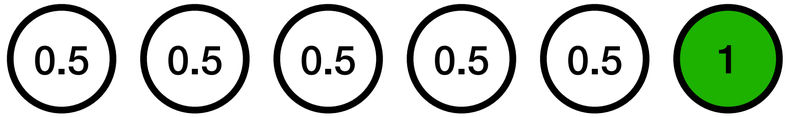
Read in the last arm

Exploit for $T-L$ rounds

BAR: best arm retention

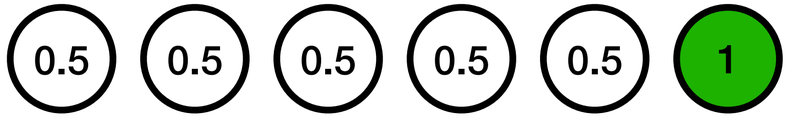
Here:Random pick two arms and pull each for $L/2$ rounds

Classic MAB Algorithm

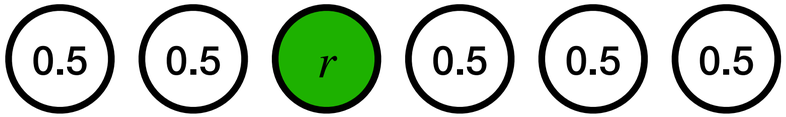


1 is the last arm

$$R_1(T) \approx L + \sqrt{2n(T-L)}$$



$r = 0.5 + \varepsilon_1$



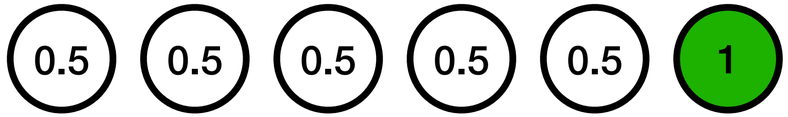
1 is the last arm

$$R_1(T) \approx L + \sqrt{2n(T-L)}$$

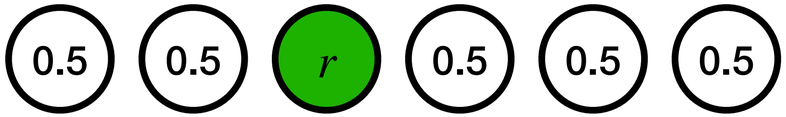
r is not the last arm

The prob. r is dropped $\approx \delta \triangleq \frac{2}{n}\exp(-L\varepsilon_1^2/8)$

$$R_2(T) \approx \varepsilon_1 \cdot (L + \delta \cdot T) + \sqrt{2n(T-L)}$$



$r = 0.5 + \varepsilon_1$



Pick $\varepsilon_1 = T^{-1/3}, L = T^{2/3}$

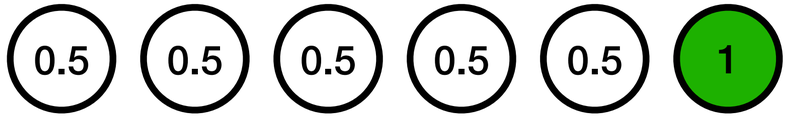
1 is the last arm

$$R_1(T) \approx L + \sqrt{2n(T-L)}$$

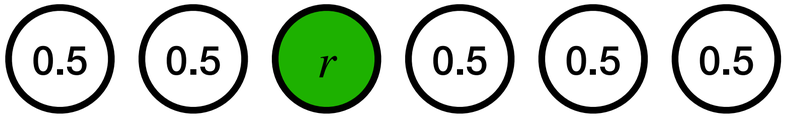
r is not the last arm

The prob. r is dropped $\approx \delta \triangleq \frac{2}{n}\exp(-L\varepsilon_1^2/8)$

$$R_2(T) \approx \varepsilon_1 \cdot (L + \delta \cdot T) + \sqrt{2n(T-L)}$$



$r = 0.5 + \varepsilon_1$



Pick $\varepsilon_1 = T^{-1/3}, L = T^{2/3}$

1 is the last arm

$$R_1(T) \approx L + \sqrt{2n(T-L)}$$

r is not the last arm

The prob. r is dropped $\approx \delta \triangleq \frac{2}{n}\exp(-L\varepsilon_1^2/8)$

$$R_2(T) \approx \varepsilon_1 \cdot (L + \delta \cdot T) + \sqrt{2n(T-L)}$$

$$R(T) \approx T^{2/3}$$

The algorithm always faces a dilemma

The algorithm always faces a dilemma

If L is large…



The algorithm always faces a dilemma

If L is large…



If L is small…



The algorithm always faces a dilemma

If L is large…



If L is small…



This is a lower bound proof

The algorithm always faces a dilemma

If L is large…



If L is small…



This is a lower bound proof

The Likelihood argument

The algorithm always faces a dilemma

If L is large…

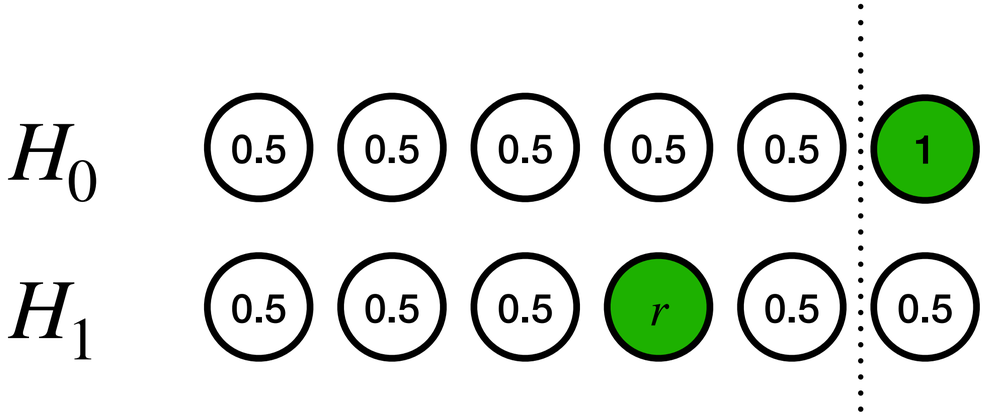


If L is small…



This is a lower bound proof

The Likelihood argument



The algorithm always faces a dilemma

If L is large…

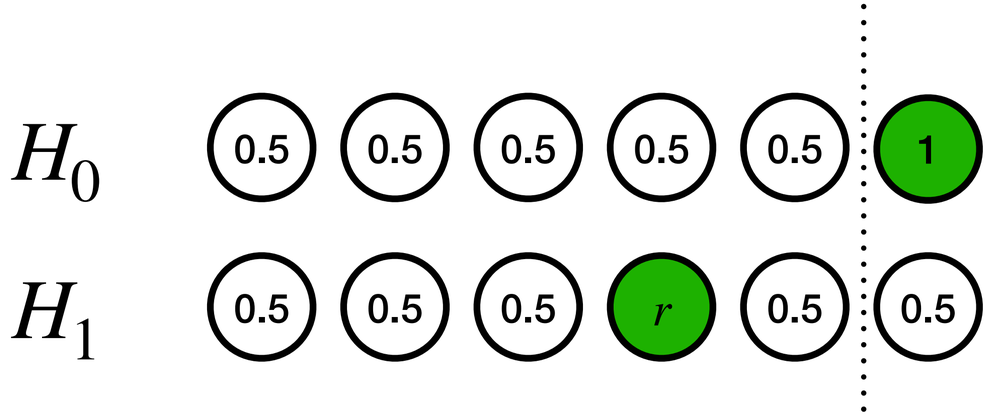


If L is small…



This is a lower bound proof

The Likelihood argument



"If the 4th arms is not pulled enough time, any algorithm $\mathscr{A}$ can not distinguish H_0 and H_1 with high probability"

Generalization to P passes

Upper Bound:

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Generalization to P passes

Upper Bound:

$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Generalization to P passes

Upper Bound:

$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Lower Bound:

"Optimal algorithm must behave like above on some instances"

“Optimal algorithm must behave like above on some instances”

“Optimal algorithm must behave like above on some instances”

Upper Bound:

$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

"Optimal algorithm must behave like above on some instances"

Define $\varepsilon_b \sim L_b^{-\frac{1}{2}}$, $r_b = 0.5 + \varepsilon_b$

Upper Bound:

$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

"Optimal algorithm must behave like above on some instances"

Upper Bound:

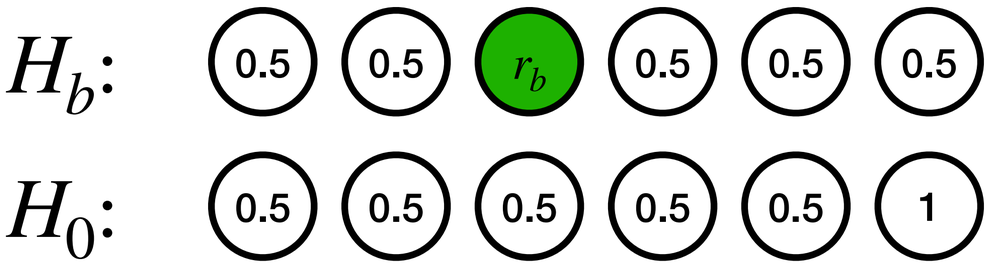
$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Define $\varepsilon_b \sim L_b^{-\frac{1}{2}}$, $r_b = 0.5 + \varepsilon_b$



"Optimal algorithm must behave like above on some instances"

Upper Bound:

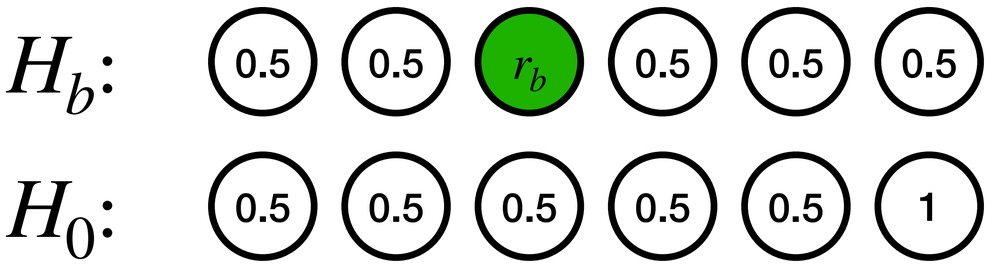
$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b-1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Define $\varepsilon_b \sim L_b^{-\frac{1}{2}}$, $r_b = 0.5 + \varepsilon_b$



Any algorithm requires L_b rounds to distinguish H_0 and H_b

"Optimal algorithm must behave like above on some instances"

Upper Bound:

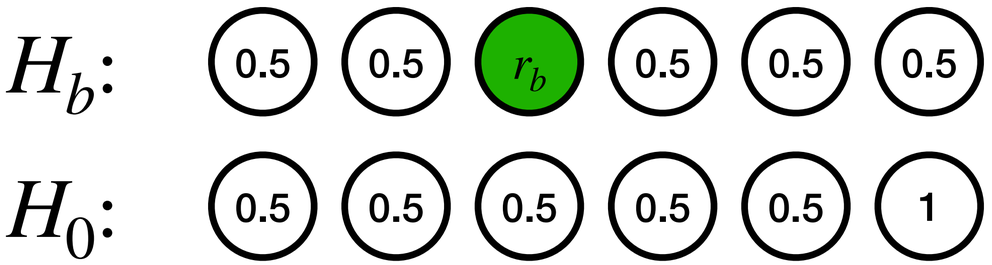
$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Define $\varepsilon_b \sim L_b^{-\frac{1}{2}}$, $r_b = 0.5 + \varepsilon_b$



Any algorithm requires L_b rounds to distinguish H_0 and H_b

Claim: Any optimal algorithm must explore $\sim L_b$ rounds on H_P at pass b

"Optimal algorithm must behave like above on some instances"

Upper Bound:

$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b - 1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Define $\varepsilon_b \sim L_b^{-\frac{1}{2}}$, $r_b = 0.5 + \varepsilon_b$

H_b: 0.5 0.5 r_b 0.5 0.5 0.5

H_0: 0.5 0.5 0.5 0.5 0.5 1

Any algorithm requires L_b rounds to distinguish H_0 and H_b

Claim: Any optimal algorithm must explore $\sim L_b$ rounds on H_P at pass b

H_P: 0.5 0.5 0.5 r_P 0.5 0.5

"Optimal algorithm must behave like above on some instances"

Upper Bound:

$L_b = T^{\frac{2^{P+1}}{2^{P+1}-1} \cdot \frac{2^b-1}{2^b}}$ is increasing

For each pass $b = 1,2,\ldots,P$

Apply BAR for L_b rounds and read in the last arm

Exploit $T - \sum_{b \in [P]} L_b$ rounds using classic MAB algorithm

Define $\varepsilon_b \sim L_b^{-\frac{1}{2}}$, $r_b = 0.5 + \varepsilon_b$

H_b: 0.5 0.5 r_b 0.5 0.5 0.5

H_0: 0.5 0.5 0.5 0.5 0.5 1

Any algorithm requires L_b rounds to distinguish H_0 and H_b

Claim: Any optimal algorithm must explore $\sim L_b$ rounds on H_P at pass b

H_P: 0.5 0.5 0.5 r_P 0.5 0.5

- If exploration in pass 1 $\gg L_1$, H_0 contributes too much regret in pass 1

Our Results

		Upper Bounds	Lower Bounds
Single Pass	Previous Results [Wang'23]	$O(n^{\frac{1}{3}} \cdot T^{\frac{2}{3}})$	$\Omega(n^{\frac{1}{3}} \cdot T^{\frac{2}{3}}), m \leq n/20$
	Our Results	$O\left(\frac{(n-m)}{n^{2/3}} \cdot T^{\frac{2}{3}}\right)$	$\tilde{\Omega}\left(\frac{(n-m)}{n^{2/3}} \cdot T^{\frac{2}{3}}\right)$

		Upper Bounds	Lower Bounds
P-Passes $\alpha = \frac{1}{2^{P+1}-1}$	Previous Results [AKP'22]	$O\left(T^{\alpha 2^P}\sqrt{nP\log T}\right)$	$\Omega\left(4^{-P}T^{\alpha 2^P}\right), m = O(n/P^2)$
	Our Results	$\tilde{O}\left(\frac{(n-m)^{1+\alpha(2^P-2)}}{n^{\alpha(2^{P+1}-2)}} \cdot T^{\alpha 2^P}\right)$	$\tilde{\Omega}\left(\frac{(n-m)^{1+\alpha(2^P-2)}}{n^{\alpha(2^{P+1}-2)}} \cdot T^{\alpha 2^P}\right)$