Bandit Learning with switching costs

Jian Ding, University of Chicago

joint with: Ofer Dekel (MSR), Tomer Koren (Technion) and Yuval Peres (MSR)

June 2016, Harvard University

Online Learning with k-Actions

player
(a.k.a. learner)

Online Learning with k-Actions

player
(a.k.a. learner)

k actions
(a.k.a. arms, experts)

Online Learning with k-Actions

player
(a.k.a. learner)

adversary
(a.k.a. environment)

Round 1

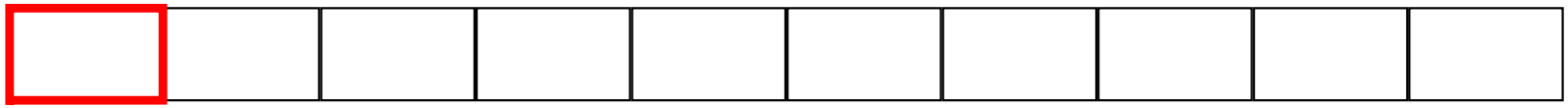
player
(a.k.a. learner)

0.9

0.2

0.6

adversary
(a.k.a. environment)



Round 1

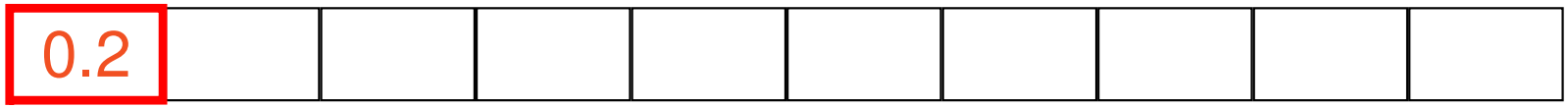
player
(a.k.a. learner)

0.9

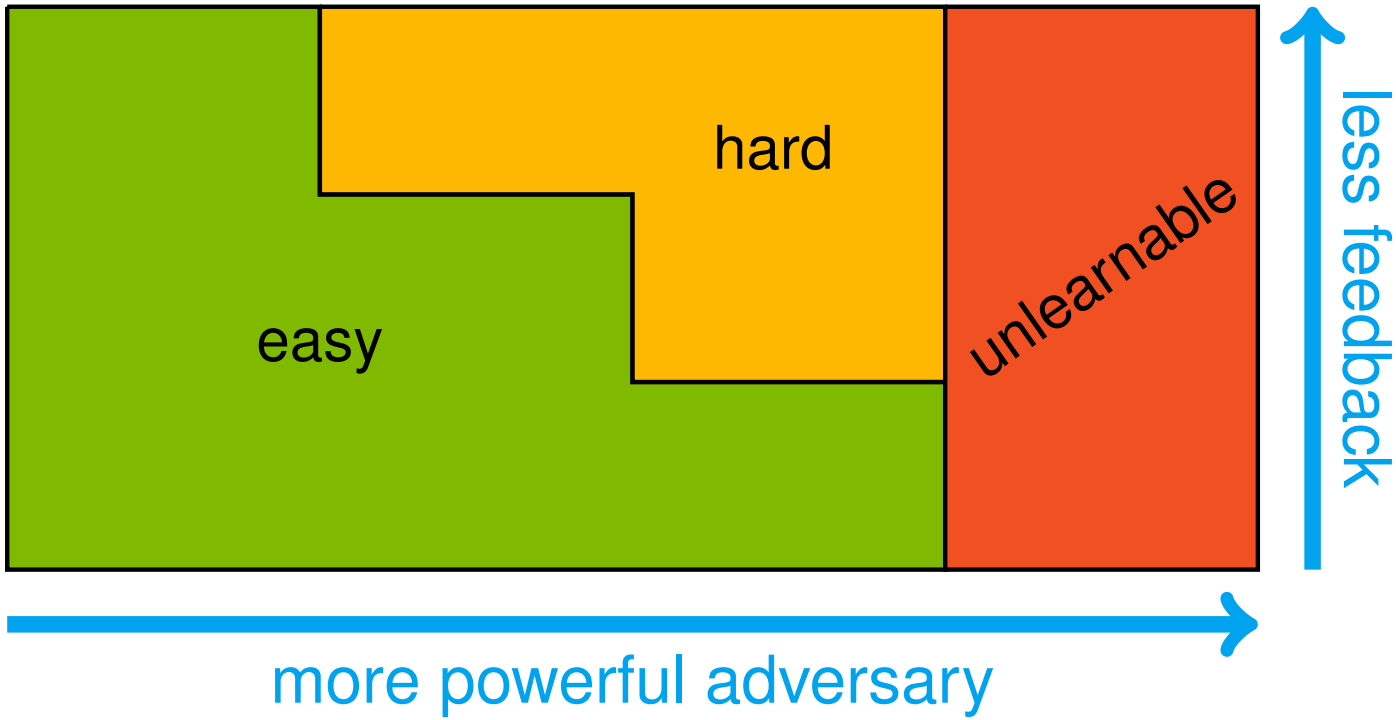
0.2

0.6

adversary
(a.k.a. environment)



Finite-Action Online Learning



- Goal: a complete characterization of “learning hardness”.

Round t

randomized player

0.1

0.7

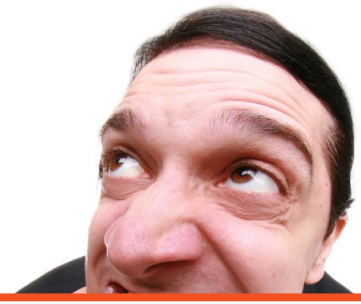
0.2

adversary

0.2	0.1	0.3	0.8	0.5					

Round t

0.1



Two Types of Adversaries

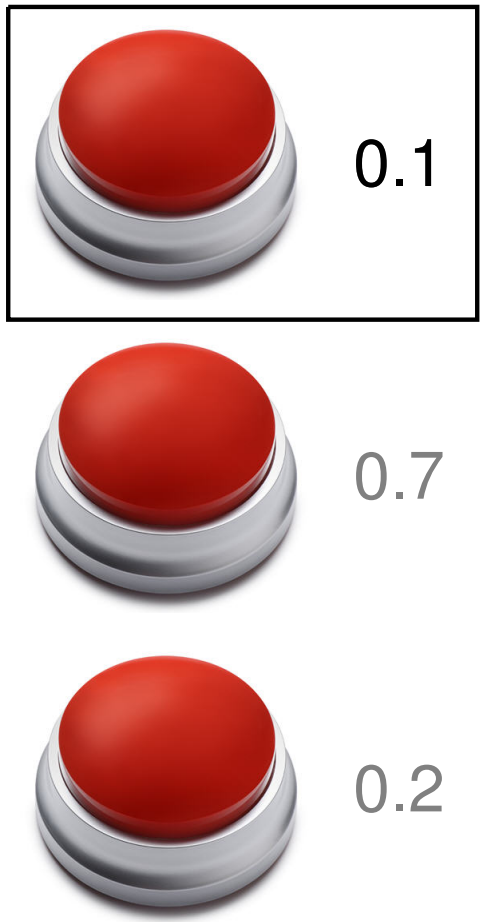
An *adaptive* adversary takes the player's past actions into account when setting loss values.

An *oblivious* adversary ignores the player's past actions when setting loss values.

0.2	0.1	0.3	0.8	0.5					

Round t

randomized player

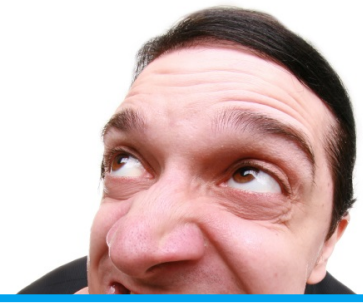


adversary

0.2	0.1	0.3	0.8	0.5	0.1				

Round t

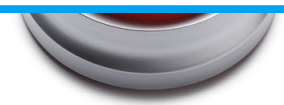
0.1



Two Feedback Models

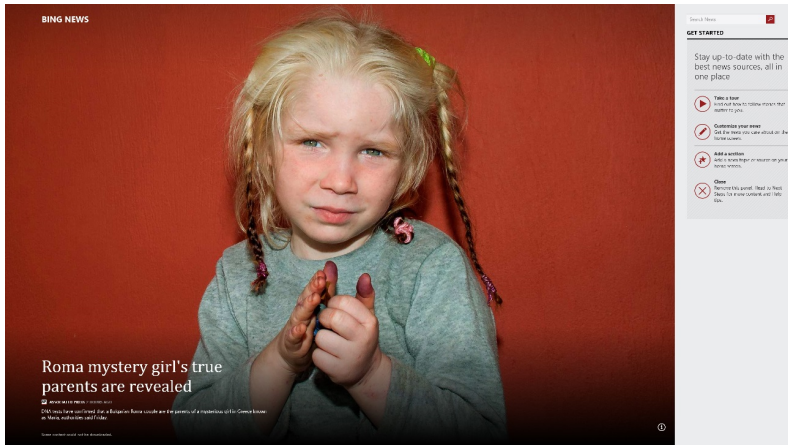
In the *bandit* feedback model, the player only sees the loss associated with his action (one number).

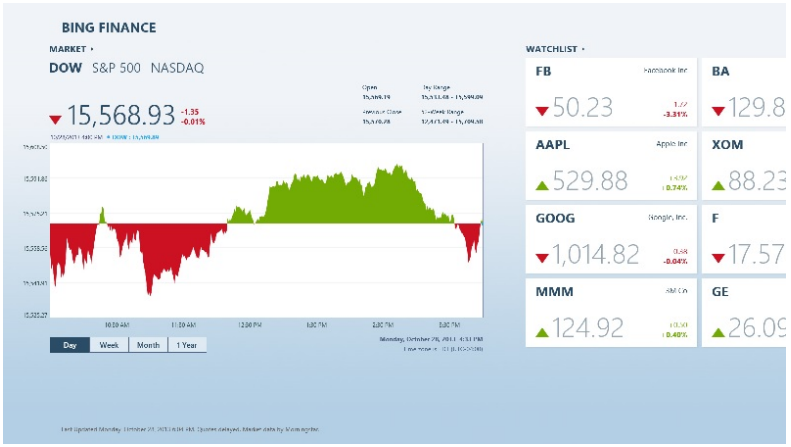
In the *full* feedback model, the player also sees the losses associated with the other actions (k numbers).



0.2	0.1	0.3	0.8	0.5	0.1				

Examples





bandit feedback

Display one of k news articles to maximize user clicks.

full feedback

Invest in one stock on each day.

More Formally

Setting A T-round repeated game between a *randomized player* and a *deterministic adaptive adversary*

Notation: player's *actions*: $\mathcal{X} = \{1, \ldots, k\}$

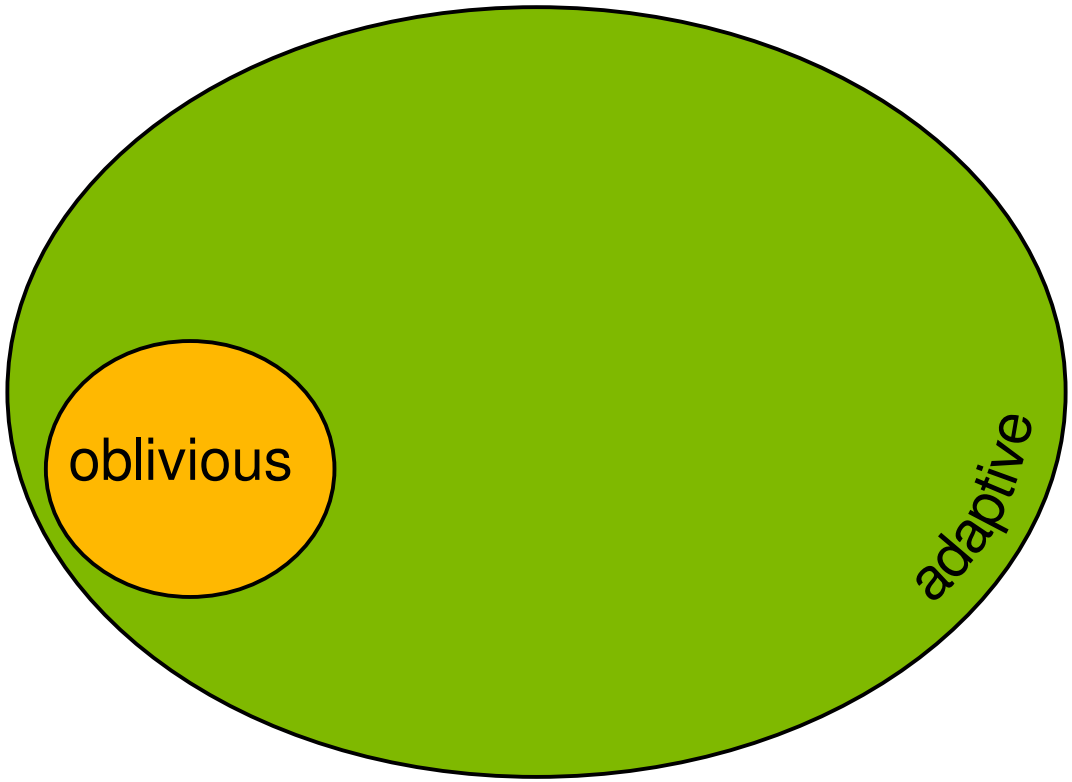
Before the game: adversary chooses sequence of *loss functions* $f_1, \ldots, f_T$, where $f_t : \mathcal{X}^t \mapsto [0, 1]$

The game: for $t = 1, \ldots, T$

- player chooses distribution μ_t over $\mathcal{X}$ and draws $X_t \sim \mu_t$
- player suffers and observes loss $f_t(X_1, \ldots, X_t)$
- if *full feedback*, player observes $\forall x \; f_t(X_1, \ldots, X_{t-1}, x)$

Adaptive vs. Oblivious

- *Adaptive*: $f_t : \mathcal{X}^t \mapsto [0, 1]$ can be any function
- *Oblivious*: adversary chooses $\ell_1, \ldots, \ell_T$, where $\ell_t : \mathcal{X} \mapsto [0, 1]$, and sets $f_t(x_1, \ldots, x_t) = \ell_t(x_t)$.



Loss, Regret

Definition Player's *expected cumulative loss* is

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(X_1, \ldots, X_t)\right] \quad .$$

Loss, Regret

Definition Player's *expected cumulative loss* is

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(X_1, \ldots, X_t)\right] \quad .$$

Definition Player's *regret* w.r.t. the best action is

$$R(T) = \mathbb{E}\left[\sum_{t=1}^{T} f_t(X_1, \ldots, X_t)\right] - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x, \ldots, x) \quad .$$

Interpretation

$R(T) = o(T) \quad \Rightarrow$ the player gets better with time.

Minimax Regret

- *Regret* measures a specific player's performance
- We want to measure the inherent difficulty of the problem

Definition The *minimax regret* $R^\star(T)$, is the inf over randomized player strategies of the sup over adversary loss sequences of the resulting expected regret.

- $R^\star(T) = \theta(\sqrt{T}) \quad \Rightarrow \quad$ problem is *easy*
- $R^\star(T) = \theta(T) \quad \Rightarrow \quad$ problem is *unlearnable*

Full + Oblivious

- a.k.a. “Predicting with Expert Advice”
- Littlestone & Warmuth (1994), Freund & Schapire (1997)

The Multiplicative Weights Algorithm

Sample X_t from μ_t where

$$\mu_t(i) \propto \exp\left(-\gamma \sum_{j=1}^{t-1} \ell_j(i)\right) .$$

Theorem $\gamma = 1/\sqrt{T}$ yields $R(T) = O(\sqrt{T \log(k)})$.

Bandit + Oblivious

- a.k.a. “The Adversarial Multiarmed Bandit Problem”
- Auer, Cesa-Bianchi, Freund, Schapire (2002)

The EXP3 Algorithm

Run the weighted majority algorithm with estimates of the full feedback vectors

$$\hat{\ell}_t(i) = \begin{cases} \frac{\ell_t(i)}{\mu_t(i)} & \text{if } i = X_t \\ 0 & \text{otherwise} \end{cases}.$$

Theorem $\mathbb{E}[\hat{\ell}_t(i)] = \ell_t(i) \quad \Rightarrow \quad R(T) = O(\sqrt{Tk})$.

Adaptive

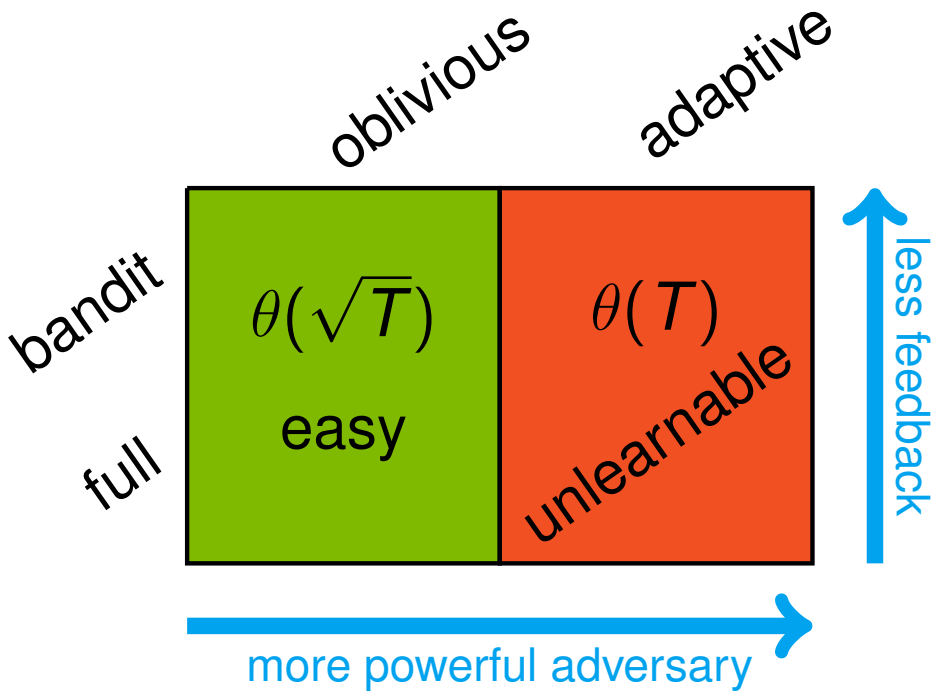
Obstacle (Arora, Dekel, Tewari 2012) $R^\star(T) = \theta(T)$ in any feedback model.

Proof w.l.o.g. assume $\mu_1(1) > 0$. Define

$$f_t(x_1, \ldots, x_t) = \begin{cases} 1 & \text{if } x_1 = 1 \\ 0 & \text{otherwise} \end{cases}.$$

This loss guarantees $R^\star(T) = \mu_1(1) \cdot T$. □

The Characterization (so far)



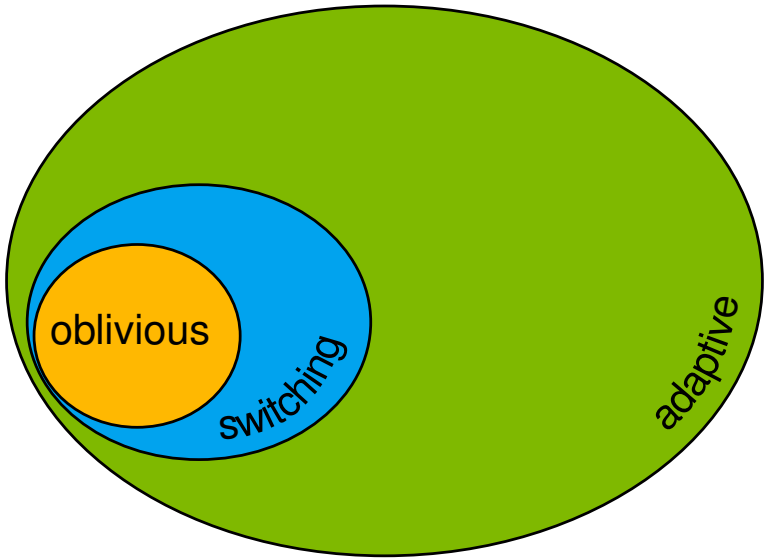
- Boring
- Feedback models seem to be equivalent (when $k = 2$, say).

Adding a Switching Cost

- The *switching cost* adversary chooses $\ell_1, \ldots, \ell_T$, where $\ell_t : \mathcal{X} \mapsto [0, 1]$, and sets

$$f_t(x_1, \ldots, x_t) = \frac{1}{2}\left(\ell_t(x_t) + \mathbb{1}_{x_t \neq x_{t-1}}\right) \quad .$$

- The "Follow the Lazy Leader" algorithm (Kalai-Vempala 05) guarantees $R(T) = O(\sqrt{T})$ (full information); also "Shrinking the dartboard" (Geulen-Vöcking-Winkler 10)



m-Memory Adversary, Counterfactual Feedback

- The *m-memory adversary* defines loss functions that depend only on the $m+1$ recent actions.

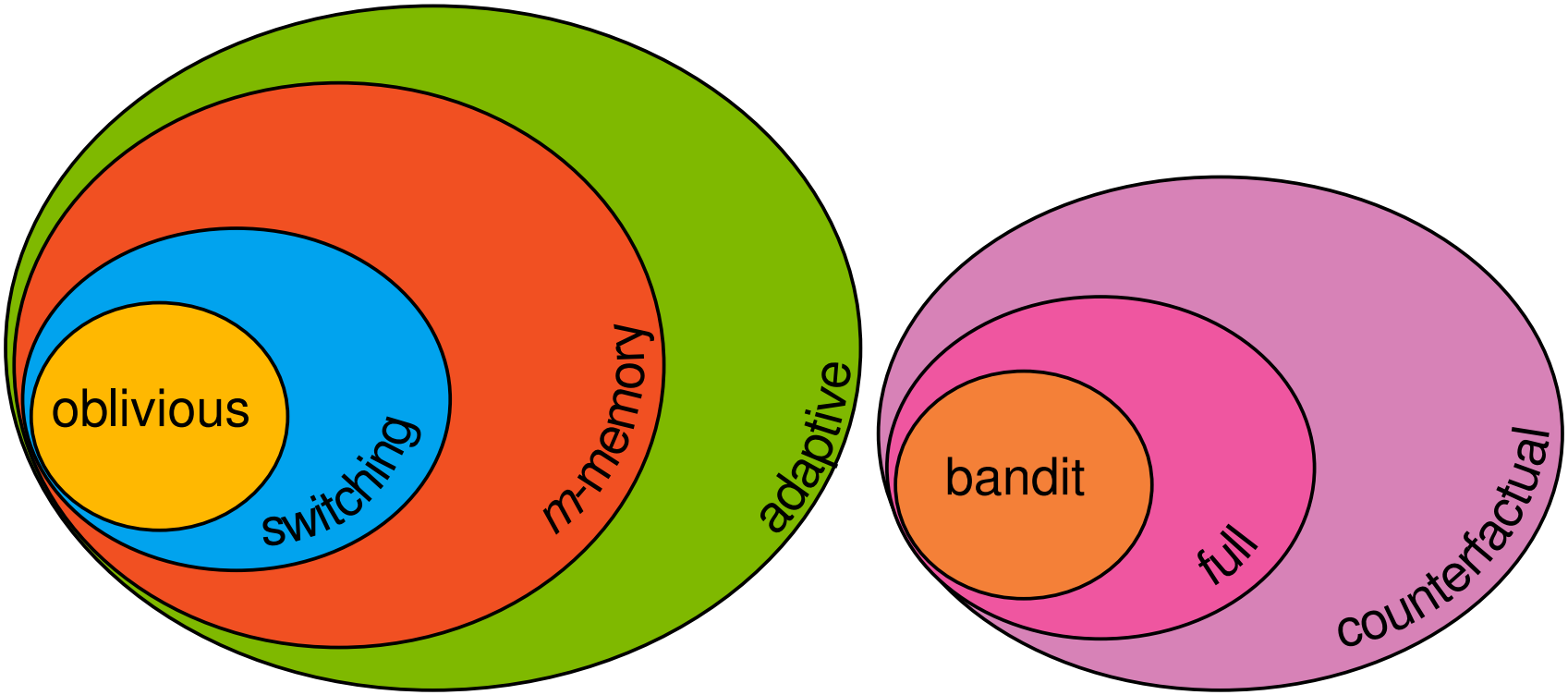
$$f_t(\underbrace{x_1, \ldots, x_t}_{t}) = f_t(\underbrace{x_{t-m}, \ldots, x_t}_{m+1}) \quad .$$

A Third Feedback Model

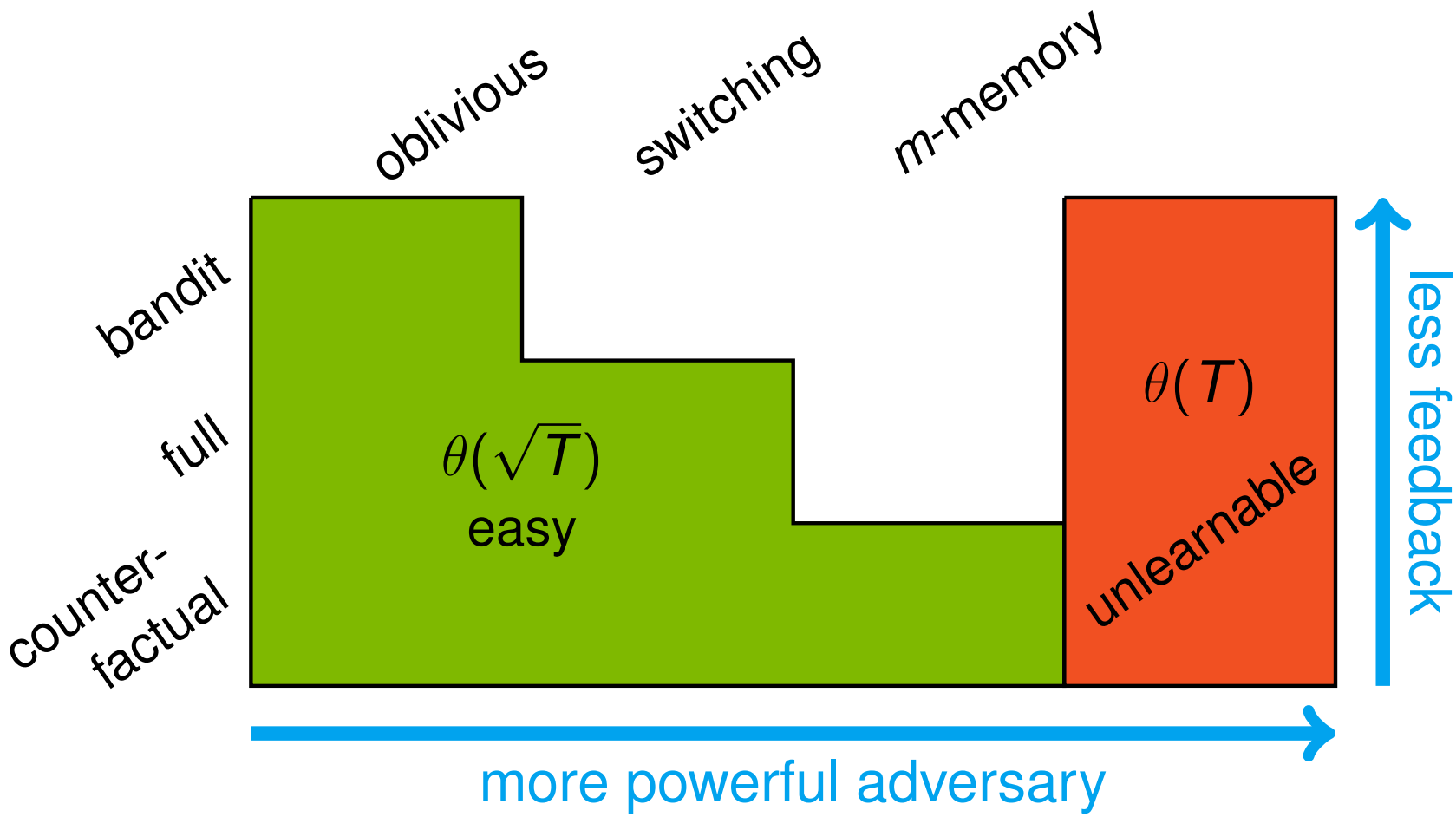
In the *counterfactual* feedback model, the player receives the entire loss function f_t.

- Merhav et al. (2002) proved $R(T) = O(T^{2/3})$.
- Gyorgy & Neu (2011) improved this to $R(T) = O(\sqrt{T})$.

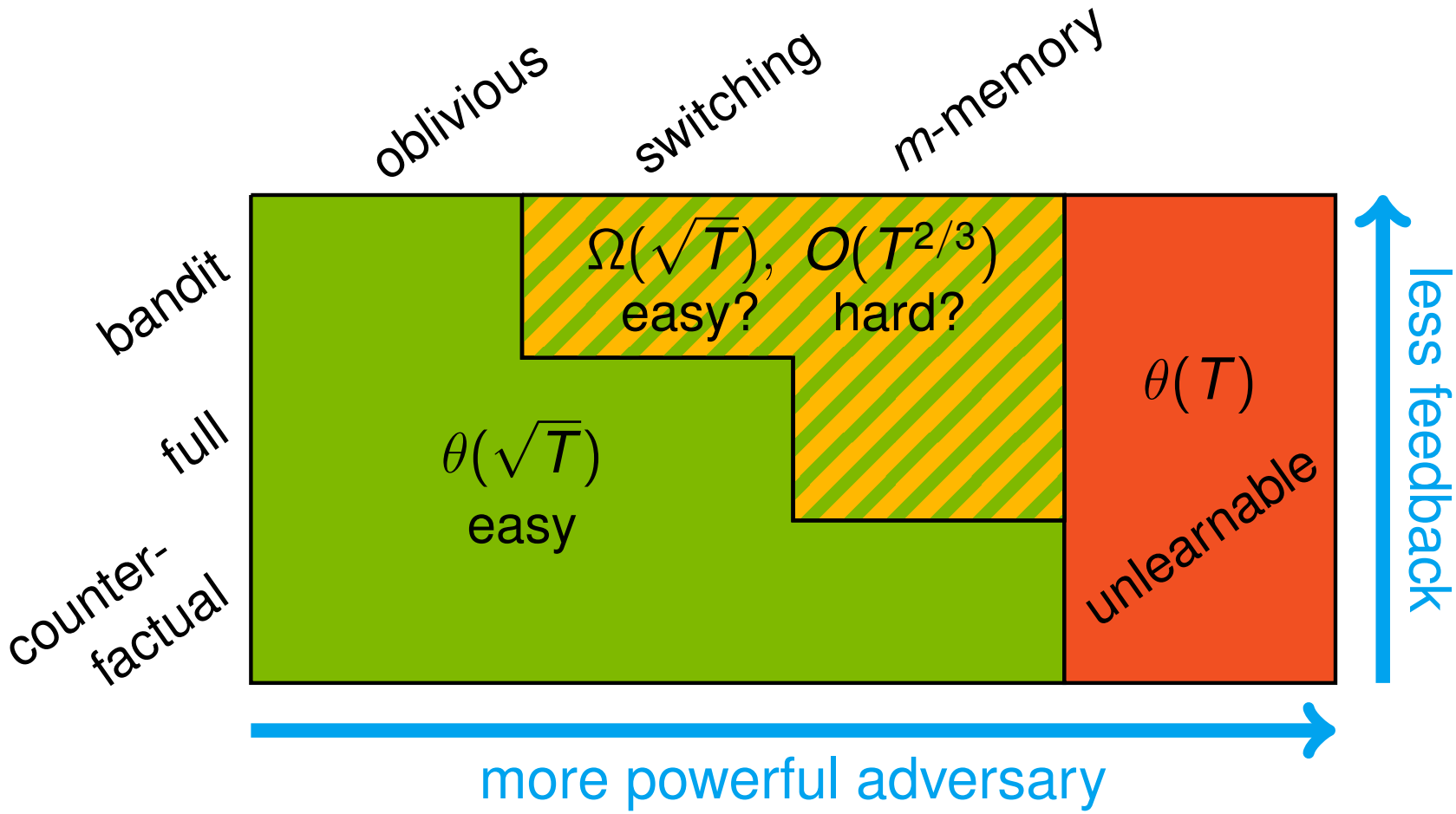
Adversaries and Feedbacks



The Characterization (so far)

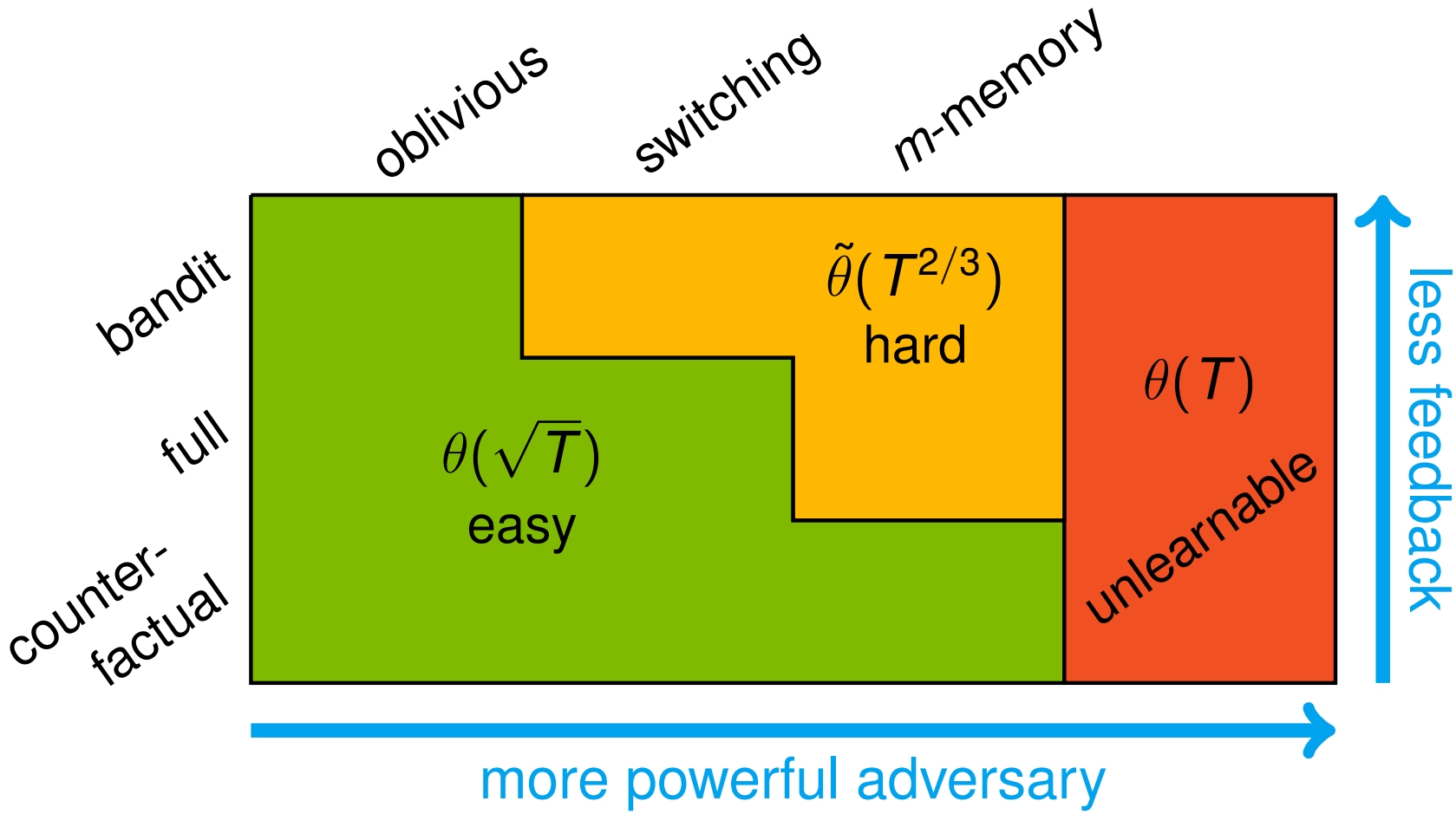


The Characterization (so far)



Arora, Dekel, Tewari 2012

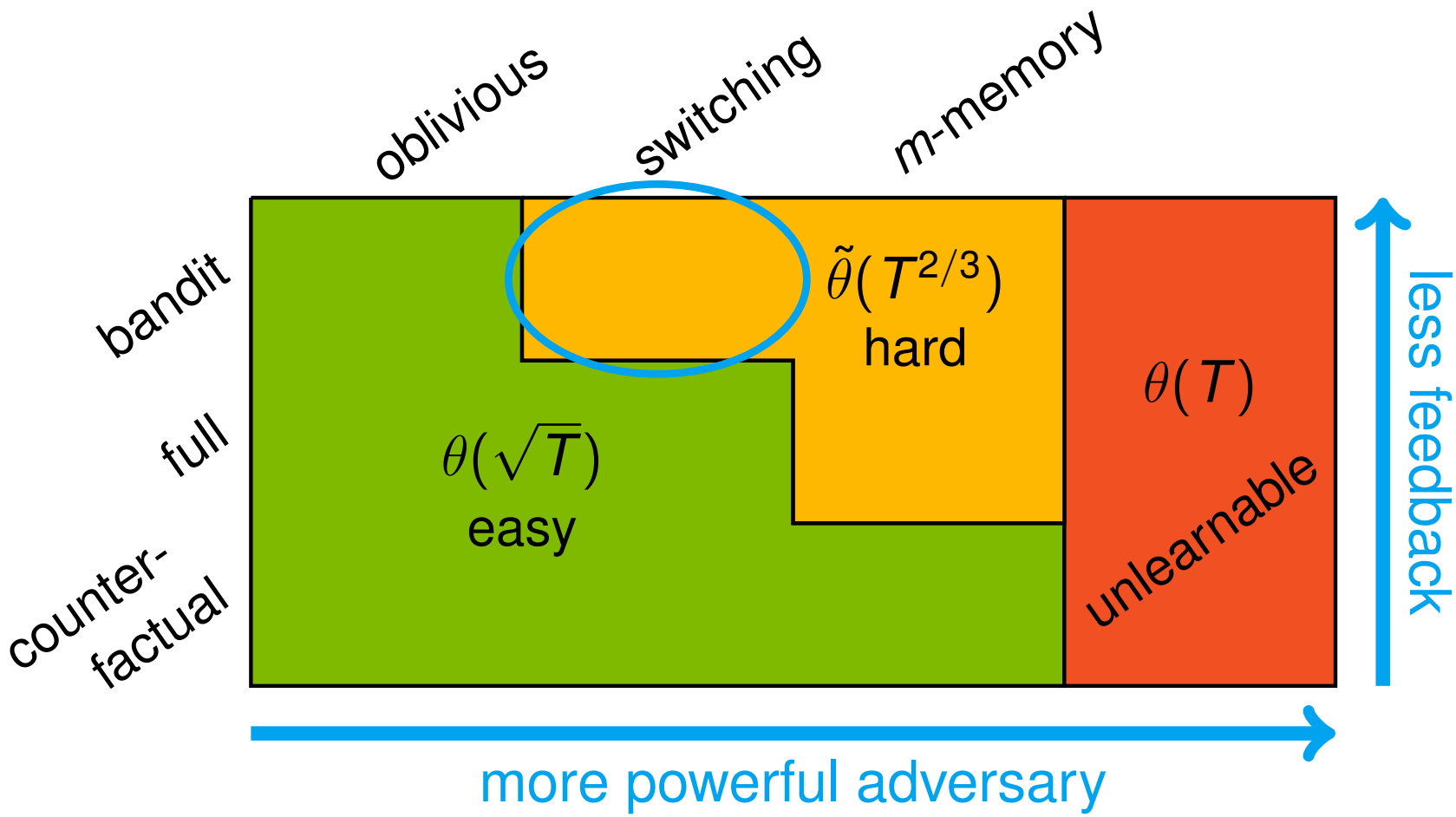
The Characterization (so far)



Cesa-Bianchi, Dekel, Shamir (2013), (Unbounded Losses)

Dekel, D., Koren, Peres (2013)

The Characterization (so far)



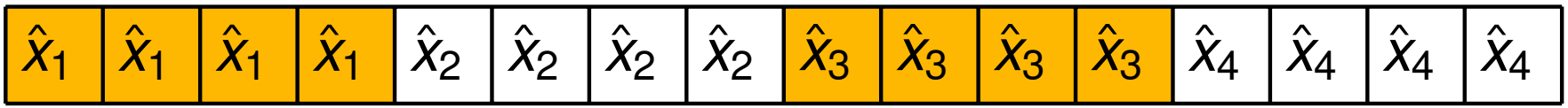
Cesa-Bianchi, Dekel, Shamir (2013), (Unbounded Losses)

Dekel, D., Koren, Peres (2013)

Bandit + Switching: Upper Bound

Algorithm

- split rounds into T/B blocks of length B.
- use EXP3 to choose action $\hat{x}_j$ for the entire block
- feedback to EXP3 is the average loss in the block



Regret Analysis

$$R(T) \leq \underbrace{T/B}_{\text{switches}} + \underbrace{B \cdot O\left(\sqrt{T/B}\right)}_{\text{loss}} = O(T/B + \sqrt{TB})$$

Minimized by selecting $B = T^{1/3}$ yielding regret $R(T) = O(T^{2/3})$.

Bandit + Switching: Lower Bound

Yao's Minimax Principle (1975) The expected regret of the best *deterministic* algorithm on a *random* loss sequence lower-bounds the expected regret of a *randomized* algorithm on the worst *deterministic* loss sequence.

Goal find a *random* loss sequence for which all *deterministic* algorithms have expected regret $\tilde{\Omega}(T^{2/3})$.

For simplicity, assume $k = 2$.

Bandit + Switching: Lower Bound

Cesa-Bianchi, Dekel, Shamir 2013: random walk construction

- Let (S_t) be a Gaussian random walk, and $\epsilon = 1/\sqrt{T}$.
- Randomly choose an action and assign to it the loss function (S_t), and the other action the loss function $(S_t + \epsilon)$.

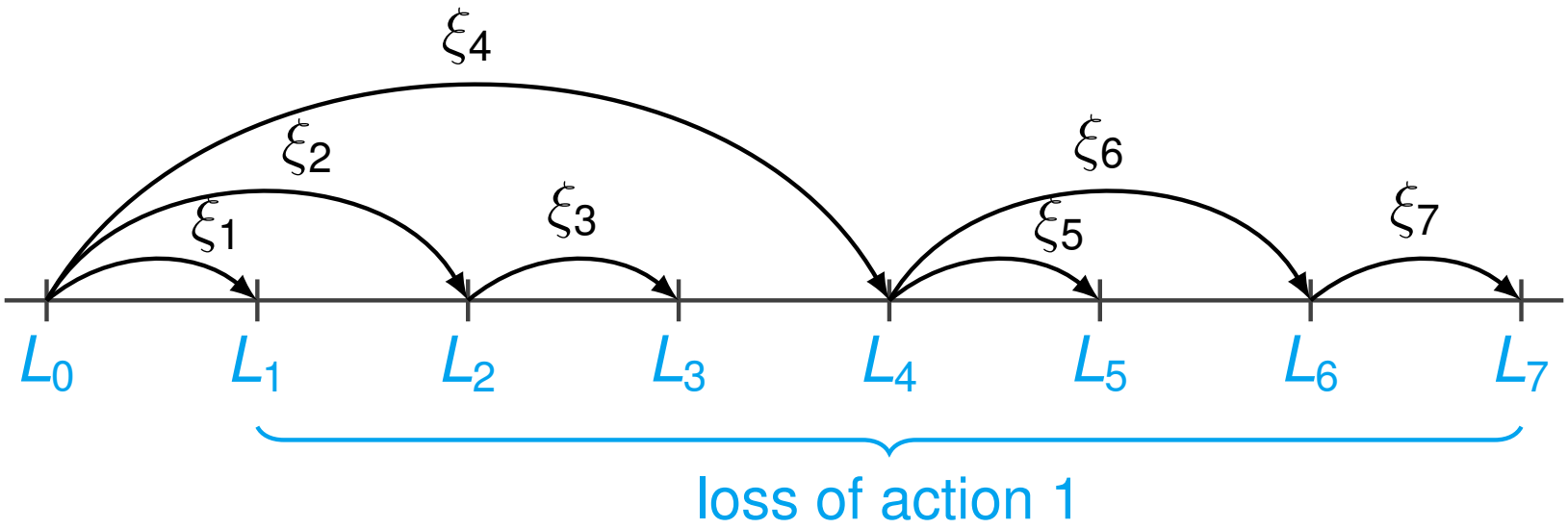
Key: $1/\epsilon^2$ switches required before determining which action is worse!

Drawback: Unbounded loss function – is hard learning an artifact of unboundedness??

Multi-Scale Random Walk

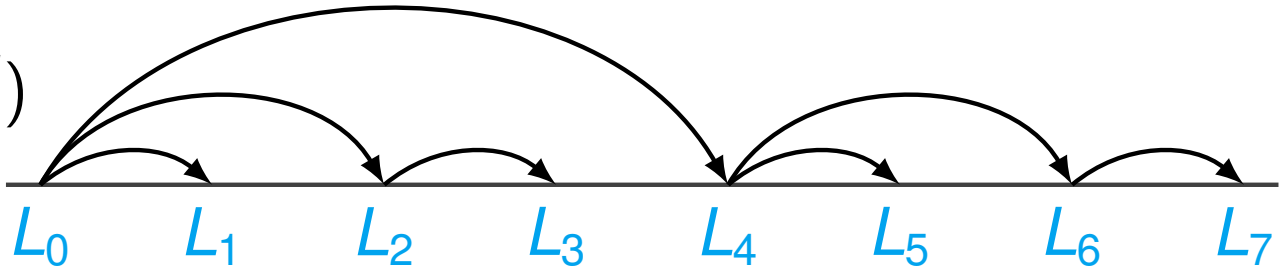
Define the loss of action 1:

- Draw independent Gaussians $\xi_1, \ldots, \xi_T \sim N(0, \sigma^2)$
- For each t, define a *parent* $\rho(t) \in \{0, \ldots, t-1\}$
- Define (recursively): $L_0 = 1/2$, $L_t = L_{\rho(t)} + \xi_t$

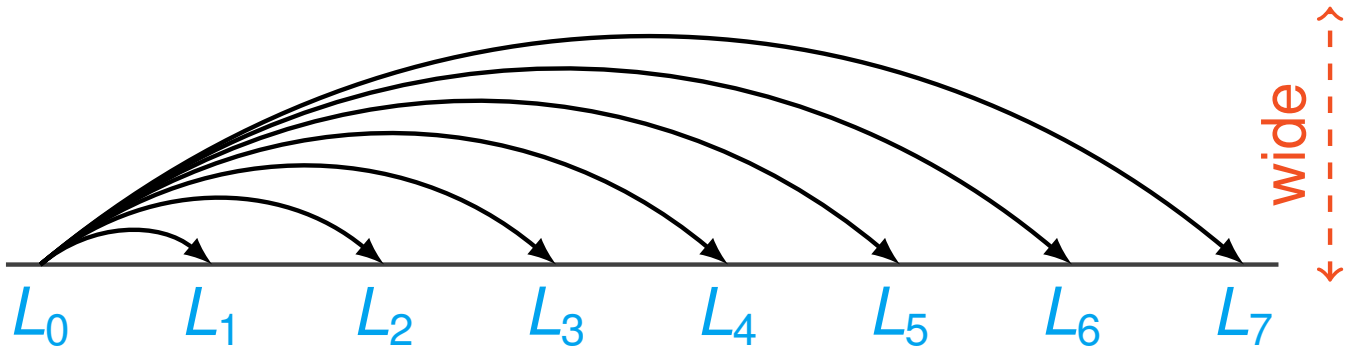


Examples

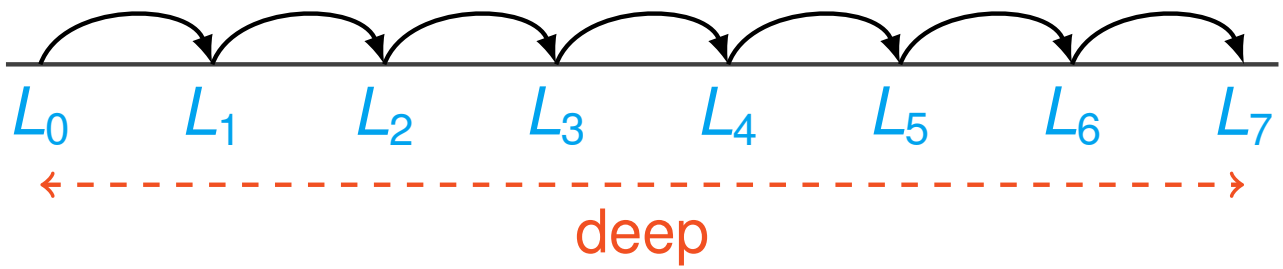
$\rho(t) = t - \gcd(t, 2^T)$



$\rho(t) = 0$



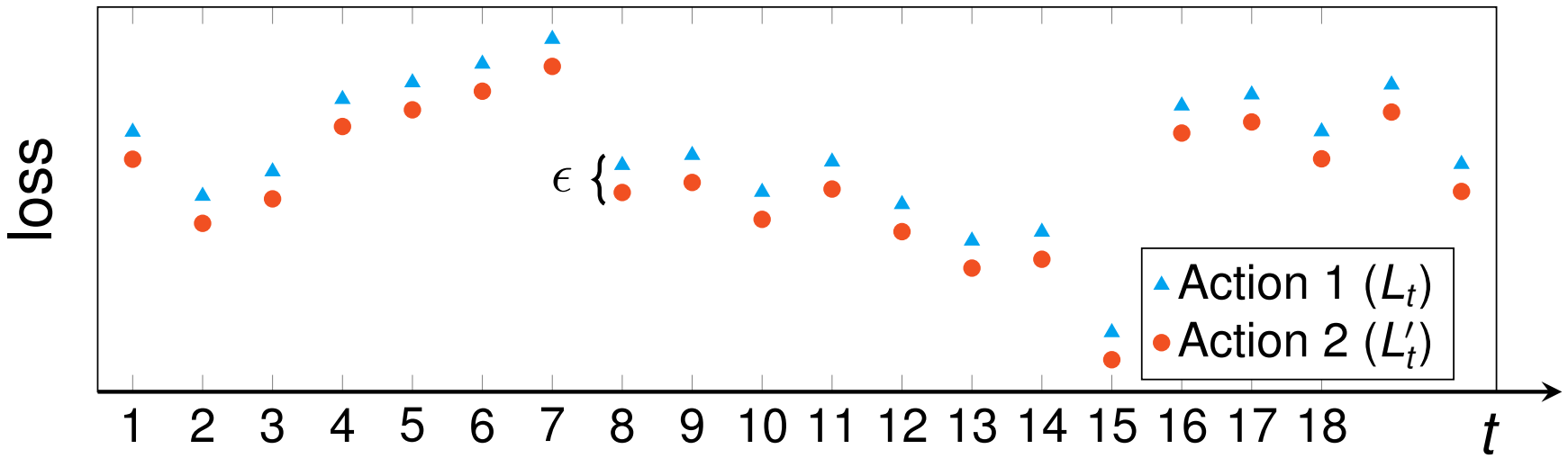
$\rho(t) = t - 1$



The Second Action

Define the loss of action 2:

- Draw a random sign χ $\quad \Big(\Pr(\chi = +1) = \Pr(\chi = -1) \Big)$
- Define $L'_t = L_t + \chi\epsilon$, where $\epsilon = T^{-1/3}$.

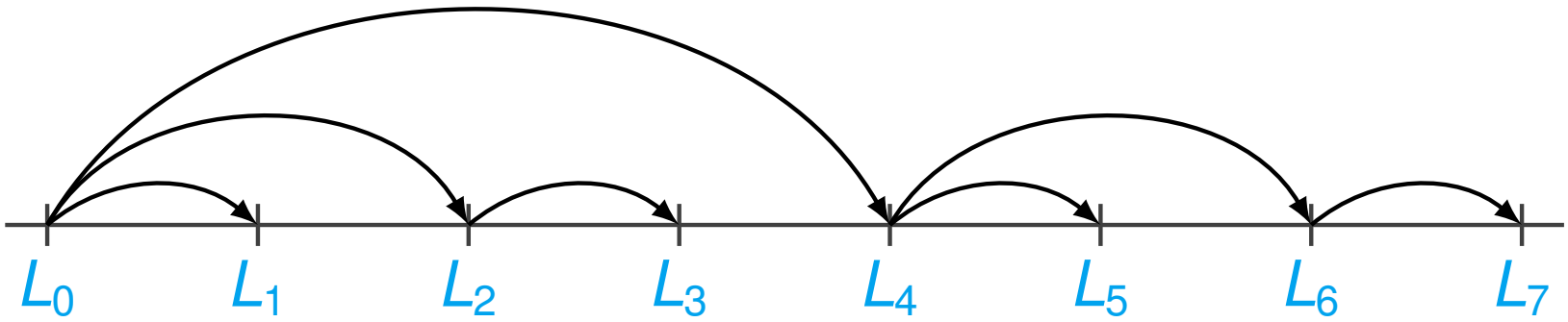


- Choose worse action $\theta(T)$ times $\Rightarrow$ $R(T) = \Omega(T^{2/3})$

The Information in One Sample

To avoid choosing the worse action $\theta(T)$ times, algorithm must identify the value of χ.

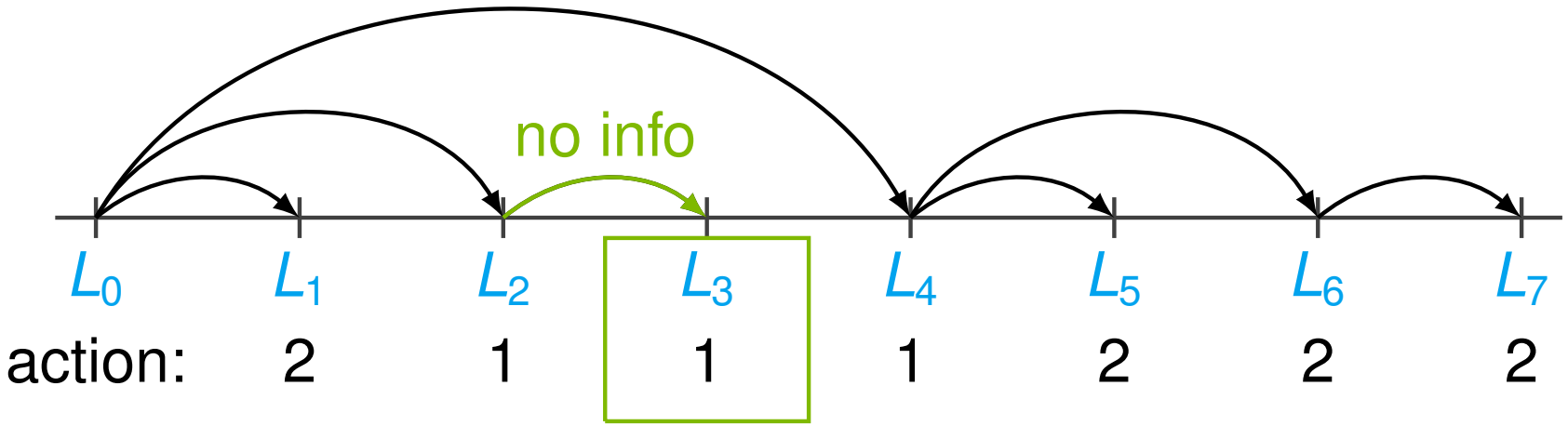
Fact **Q:** How many samples to estimate the mean of a Gaussian with accuracy ϵ? **A:** $\left(\frac{\sigma}{\epsilon}\right)^2$



The Information in One Sample

To avoid choosing the worse action $\theta(T)$ times, algorithm must identify the value of χ.

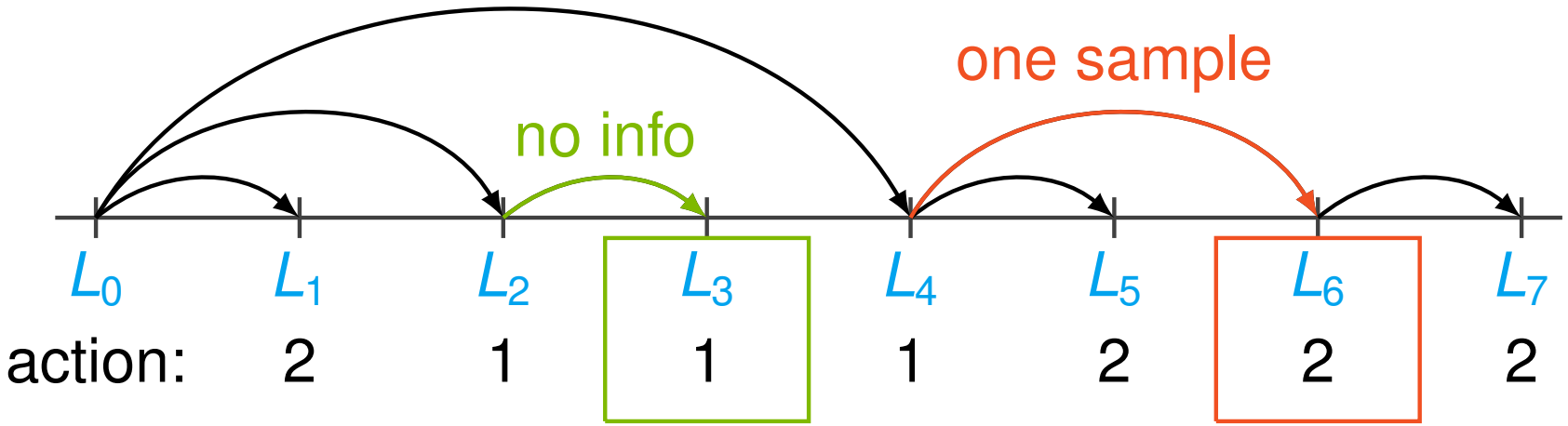
Fact **Q:** How many samples to estimate the mean of a Gaussian with accuracy ϵ? **A:** $\left(\frac{\sigma}{\epsilon}\right)^2$



The Information in One Sample

To avoid choosing the worse action $\theta(T)$ times, algorithm must identify the value of χ.

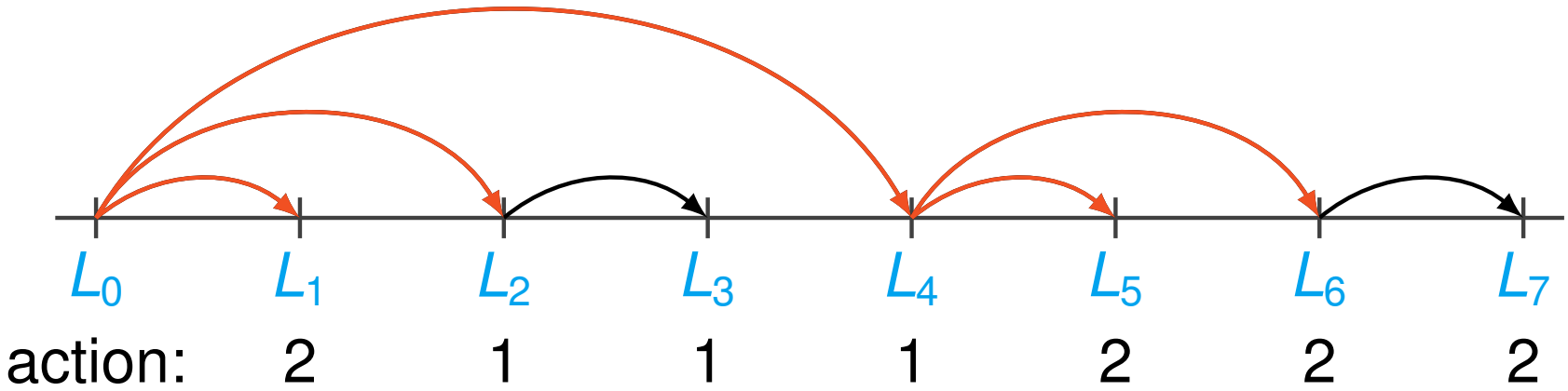
Fact **Q:** How many samples to estimate the mean of a Gaussian with accuracy ϵ? **A:** $\left(\frac{\sigma}{\epsilon}\right)^2$



The Information in One Sample

To avoid choosing the worse action $\theta(T)$ times, algorithm must identify the value of χ.

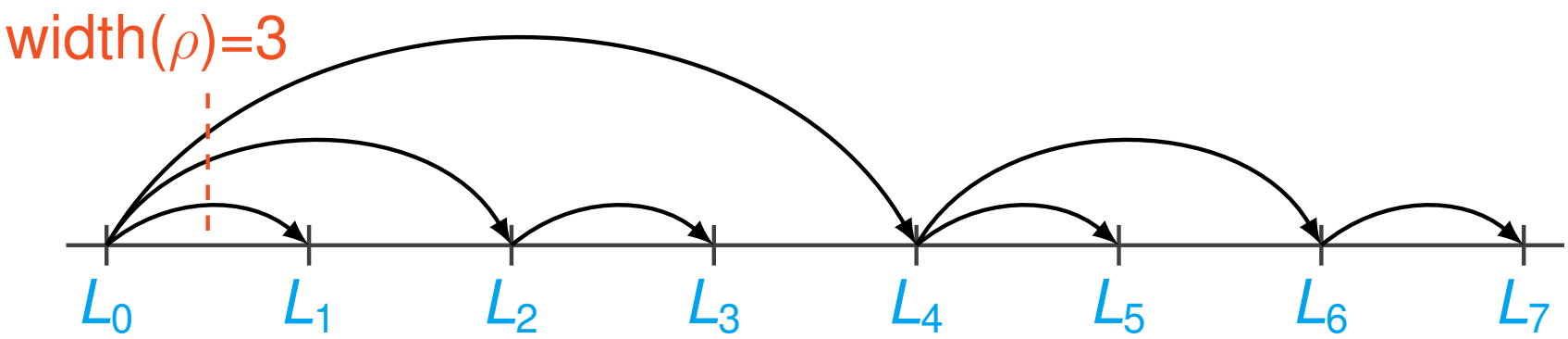
Fact **Q:** How many samples to estimate the mean of a Gaussian with accuracy ϵ? **A:** $\left(\frac{\sigma}{\epsilon}\right)^2$



How many red edges? Player needs at least $\sigma^2 T^{2/3}$

Counting the Information

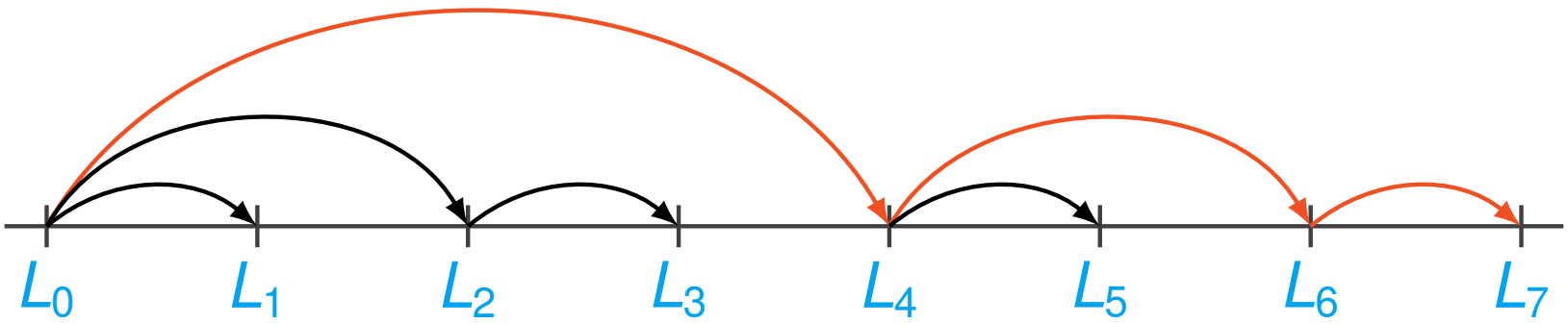
Define width(ρ) is the maximum size of any vertical cut in the graph induced by ρ.



Lemma A switch contributes $\leq$ width(ρ) samples.

Depth

Define depth(ρ) is the length of the longest path.



Loss should remain bounded in $[0, 1]$ $\Rightarrow$ set $\sigma \sim \frac{1}{\text{depth}(\rho)}$.

Putting it All Together

- $\sigma^2 T^{2/3}$ samples needed
- each switch gives $\leq$ width(ρ) samples
- loss bounded in $[0, 1] \quad \Rightarrow \quad \sigma^2 \sim \frac{1}{\text{depth}(\rho)}$.

Conclusion Number of switches needed to determine the better action $\sim \frac{T^{2/3}}{\text{width}(\rho)\cdot\text{depth}(\rho)^2}$

Putting it All Together

- $\sigma^2 T^{2/3}$ samples needed
- each switch gives $\leq$ width(ρ) samples
- loss bounded in $[0, 1] \quad \Rightarrow \quad \sigma^2 \sim \frac{1}{\text{depth}(\rho)}$.

Conclusion Number of switches needed to determine the better action $\sim \frac{T^{2/3}}{\text{width}(\rho)\cdot\text{depth}(\rho)^2}$

Choose $\rho(t) = t - \gcd(t, 2^T)$

Lemma depth(ρ) $\leq \log(T)$ and width(ρ) $\leq \log(T)+1$

Corollaries & Extensions

Corollary Exploration requires switching. e.g., EXP3 switches $\theta(T)$ times.

Dependence on k The minimax regret of the multi-armed bandit with switching costs is $\tilde{\theta}(T^{2/3}k^{1/3})$

Implications on other models The minimax regret of learning an adversarial deterministic MDP is $\tilde{\theta}(T^{2/3})$

Summary

- A complete characterization of “learning hardness”.
- There exist online learning problems that are hard yet learnable.
- Learning with bandit feedback can be strictly harder than learning with full feedback.
- Exploration requires extensive switching.

The End