Trade-offs in Nonstochastic Bandits

Nicolò Cesa-Bianchi

Università degli Studi di Milano
The nonstochastic bandit problem

- Initially studied as a repeated unknown game [Baños, 1968]
The nonstochastic bandit problem

- Initially studied as a **repeated unknown game** [Baños, 1968]
- ML/CS contribution: simple algorithms, tight analysis, extensions and applications
The nonstochastic bandit problem

- Initially studied as a repeated unknown game \[ \text{[Baños, 1968]} \]
- ML/CS contribution: simple algorithms, tight analysis, extensions and applications
- A great setting for the study of learning with partial and delayed feedback
The nonstochastic bandit problem

- Initially studied as a repeated unknown game [Baños, 1968]
- ML/CS contribution: simple algorithms, tight analysis, extensions and applications
- A great setting for the study of learning with partial and delayed feedback
- Nonstochastic setting $\rightarrow$ regret minimization $\rightarrow$ simple and elegant algorithms
The nonstochastic bandit problem

A sequential decision problem

- **K** actions
- Unknown **deterministic** assignment of losses to actions
  \[ \ell_t = (\ell_t(1), \ldots, \ell_t(K)) \in [0, 1]^K \text{ for } t = 1, 2, \ldots \]
The nonstochastic bandit problem

A sequential decision problem

- $K$ actions
- Unknown **deterministic** assignment of losses to actions
  \[ \ell_t = (\ell_t(1), \ldots, \ell_t(K)) \in [0, 1]^K \text{ for } t = 1, 2, \ldots \]

For $t = 1, 2, \ldots$

1. Player picks an action $I_t$ (possibly using randomization) and incurs loss $\ell_t(I_t)$
The nonstochastic bandit problem

A sequential decision problem

- \( K \) actions
- Unknown **deterministic** assignment of losses to actions
  \[ \ell_t = (\ell_t(1), \ldots, \ell_t(K)) \in [0, 1]^K \text{ for } t = 1, 2, \ldots \]

For \( t = 1, 2, \ldots \)

1. Player picks an action \( I_t \) (possibly using randomization) and incurs loss \( \ell_t(I_t) \)
2. Player gets feedback information
The nonstochastic bandit problem

A sequential decision problem

- **K** actions
- Unknown **deterministic** assignment of losses to actions
  \[ \ell_t = (\ell_t(1), \ldots, \ell_t(K)) \in [0, 1]^K \text{ for } t = 1, 2, \ldots \]

For \( t = 1, 2, \ldots \)

1. Player picks an action \( I_t \) (possibly using randomization) and incurs loss \( \ell_t(I_t) \)
2. Player gets feedback information
   - **Bandit feedback**: Only \( \ell_t(I_t) \) is revealed
The nonstochastic bandit problem

A sequential decision problem

- K actions
- Unknown **deterministic** assignment of losses to actions
  \[ \ell_t = (\ell_t(1), \ldots, \ell_t(K)) \in [0, 1]^K \text{ for } t = 1, 2, \ldots \]

For \( t = 1, 2, \ldots \)

1. Player picks an action \( I_t \) (possibly using randomization) and incurs loss \( \ell_t(I_t) \)
2. Player gets feedback information
   - **Bandit feedback**: Only \( \ell_t(I_t) \) is revealed
   - **Expert feedback**: The entire loss vector \( \ell_t \) is revealed
Regret of randomized agent $I_1, I_2, \ldots$

\[
R_T \overset{\text{def}}{=} \mathbb{E} \left[ \sum_{t=1}^{T} \ell_t(I_t) \right] - \min_{i=1, \ldots, K} \sum_{t=1}^{T} \ell_t(i) \overset{\text{want}}{=} o(T)
\]
Regret of randomized agent $I_1, I_2, \ldots$

\[
R_T \overset{\text{def}}{=} \mathbb{E} \left[ \sum_{t=1}^{T} \ell_t(I_t) \right] - \min_{i=1, \ldots, K} \sum_{t=1}^{T} \ell_t(i) = o(T)
\]

Minimax rates

- **Experts:** $R_T = \Theta(\sqrt{T \ln K})$
- **Bandits:** $R_T = \Theta(\sqrt{TK})$
Summary

1. A brief digression

2. The silver bullet

3. The space tradeoff

4. The time tradeoff
A brief digression

The silver bullet

The space tradeoff

The time tradeoff
A brief history of (nonstochastic) bandits
A brief history of (nonstochastic) bandits
How to bore your audience to death
In 1994 Peter Auer started his civilian service as conscientious objector in Graz.
A brief history of (nonstochastic) bandits
How to bore your audience to death

- In 1994 Peter Auer started his civilian service as conscientious objector in Graz
In 1994 Peter Auer started his civilian service as conscientious objector in Graz.

His position became vacant and I moved to Graz to take it.
In 1994 Peter Auer started his civilian service as conscientious objector in Graz.

His position became vacant and I moved to Graz to take it.

One day Peter went to a conference and Rob Schapire introduced him to the problem.
In 1994 Peter Auer started his civilian service as conscientious objector in Graz.

His position became vacant and I moved to Graz to take it.

One day Peter went to a conference and Rob Schapire introduced him to the problem.

Back from the conference, Peter and I started working on an algorithm.
In 1994 Peter Auer started his civilian service as conscientious objector in Graz.

His position became vacant and I moved to Graz to take it.

One day Peter went to a conference and Rob Schapire introduced him to the problem.

Back from the conference, Peter and I started working on an algorithm.

We proved a regret bound of $O(T^{4/5})$ and sent the paper to STOC.
A brief history of (nonstochastic) bandits

Worst-Case Analysis of the Bandit Problem

Peter Auer
IGI, Technische Universität Graz
Klosterwiegasse 32/2
A-8010 Graz, Austria.
pauer@igi.tu-graz.ac.at

Nicolò Cesa-Bianchi
DSI, University of Milan
Via Comelico 39
I-20135 Milano, Italy.
cesabian@dsi.unimi.it

November 30, 1994

Abstract

The multi-armed bandit is a classical problem in the area of sequential decision theory and has been studied under a variety of statistical assumptions. In this work we investigate the bandit problem from a purely worst-case standpoint. We present a randomized algorithm with an expected total reward of $G - O(G^{4/5}K^{6/5})$ (disregarding log factors), where $K$ is the number of arms and $G$ is the (unknown) total reward of the best arm. Our analysis holds with no assumptions whatsoever on the way rewards are generated, other than being independent of the algorithm’s randomization. Our results can also be interpreted as a novel extension of the on-line prediction model, an intensively studied framework in learning theory.
Rob Schapire was also working on a bandit algorithm with Yoav Freund, but they were too late for STOC...
Rob Schapire was also working on a bandit algorithm with Yoav Freund, but they were too late for STOC...

They learned about our paper and asked us to withdraw from STOC and join forces for a FOCS submission.
Rob Schapire was also working on a bandit algorithm with Yoav Freund, but they were too late for STOC...

They learned about our paper and asked us to withdraw from STOC and join forces for a FOCS submission

Well, their algorithm had a much better (though still suboptimal) $T^{2/3}$ rate...
Gambling in a rigged casino:
The adversarial multi-armed bandit problem

Peter Auer
Computer & Information Sciences
University of California
Santa Cruz, CA 95064
pauer@cse.ucsc.edu

Nicolò Cesa-Bianchi
DSI
Università di Milano
20135 Milano (Italy)
cesabian@dsi.unimi.it

Yoav Freund
AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974
{yoav, schapire}@research.att.com

Robert E. Schapire

Abstract

In the multi-armed bandit problem, a gambler must decide which arm of $K$ non-identical slot machines to play in a sequence of trials so as to maximize his reward. This classical problem has received much attention because of the simple model it provides of the trade-off between exploration (trying out each arm to find the best one) and exploitation (playing the arm believed to give the best payoff). Past solutions for the bandit problem have almost always relied on assumptions about the statistics of the slot machines.

In this work, we make no statistical assumptions whatsoever about the nature of the process generating the payoffs of the slot machines. We give a solution to the bandit problem in which an adversary, rather than a well-behaved stochastic process, has complete control over the payoffs. In a sequence of $T$ plays, we prove that the expected per-round payoff of our algorithm approaches that of the best arm at the rate $O(T^{-1/3})$, and we give an improved rate of convergence when the best arm has fairly low payoff.

best (“exploitation”), he may fail to discover that one of the other arms actually has a higher average return. On the other hand, if he spends too much time trying out all the machines and gathering statistics (“exploration”), he may fail to play the best arm often enough to get a high total return.

As a more practically motivated example, consider the task of repeatedly choosing a route for transmitting packets between two points in a communication network. Suppose there are $K$ possible routes and the transmission cost is reported back to the sender. Then the problem can be seen as that of selecting a route for each packet so that the total cost of transmitting a large set of packets would not be much larger than the cost incurred by sending them all on the single best route.

In the past, the bandit problem has almost always been studied with the aid of statistical assumptions on the process generating the rewards for each arm. In the gambling example, for instance, it might be natural to assume that the distribution of rewards for each arm is Gaussian and time-invariant. However, it is likely that the rates associated with each route in
A brief history of (nonstochastic) bandits

- The paper appeared in FOCS 1995, and shortly after we figured out how to get the correct $\sqrt{T}$ regret rate.
The paper appeared in FOCS 1995, and shortly after we figured out how to get the correct $\sqrt{T}$ regret rate.

We kept working adding more results.
A brief history of (nonstochastic) bandits

- The paper appeared in FOCS 1995, and shortly after we figured out how to get the correct $\sqrt{T}$ regret rate
- We kept working adding more results
- In 1998 we sent it to JACM
A brief history of (nonstochastic) bandits

- The paper appeared in FOCS 1995, and shortly after we figured out how to get the correct $\sqrt{T}$ regret rate
- We kept working adding more results
- In 1998 we sent it to JACM

Gambling in a rigged casino:
The adversarial multi-armed bandit problem*

Peter Auer
Institute for Theoretical Computer Science
University of Technology Graz
A-8010 Graz (Austria)
pauer@igi.tu-graz.ac.at

Nicolò Cesa-Bianchi
Department of Computer Science
Università di Milano
I-20135 Milano (Italy)
cesabian@dsi.unimi.it

Yoav Freund
AT&T Labs
180 Park Avenue
Florham Park, NJ 07932-0971
{yoav, schapire}@research.att.com

Internal Report 223-98
DSI, Università di Milano, Italy
A brief history of (nonstochastic) bandits

- JACM promptly rejected the paper
  (sorry guys, but these do not like bandits to us!)
A brief history of (nonstochastic) bandits

- JACM promptly rejected the paper (sorry guys, but these do not like bandits to us!)
- We re-submitted to SICOMP in 2001 and got quickly accepted
A brief history of (nonstochastic) bandits

- JACM promptly rejected the paper
  (sorry guys, but these do not like bandits to us!)
- We re-submitted to SICOMP in 2001 and got quickly accepted

20 Most Read Articles

- The Nonstochastic Multiarmed Bandit Problem
- Quantum Complexity Theory
- Depth-First Search and Linear Graph Algorithms
- Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer
- The Complexity of Computing a Nash Equilibrium
The silver bullet
The Hedge/Exp3 algorithm

Agent’s strategy

\[ P_t(I_t = i) \propto \exp \left( -\eta \sum_{s=1}^{t-1} \hat{\ell}_s(i) \right) \quad i = 1, \ldots, N \]
The Hedge/Exp3 algorithm

Agent’s strategy

\[ P_t(I_t = i) \propto \exp \left( -\eta \sum_{s=1}^{t-1} \hat{\ell}_s(i) \right) \quad i = 1, \ldots, N \]

\[ \hat{\ell}_t(i) = \begin{cases} \frac{\ell_t(i)}{P_t(\ell_t(i) \text{ observed})} & \text{if } \ell_t(i) \text{ is observed} \\ 0 & \text{otherwise} \end{cases} \]
The Hedge/Exp3 algorithm

Agent’s strategy

- \( \mathbb{P}_t(I_t = i) \propto \exp\left(-\eta \sum_{s=1}^{t-1} \hat{\ell}_s(i)\right) \quad i = 1, \ldots, N \)
- \( \hat{\ell}_t(i) = \begin{cases} \frac{\ell_t(i)}{\mathbb{P}_t(\ell_t(i) \text{ observed})} & \text{if } \ell_t(i) \text{ is observed} \\ 0 & \text{otherwise} \end{cases} \)

Experts: \( \hat{\ell}_t = \ell_t \) (Hedge)

Bandits: Only one non-zero component in \( \hat{\ell}_t \) (Exp3)
The Hedge/Exp3 algorithm

Agent’s strategy

- \( P_t(I_t = i) \propto \exp \left( -\eta \sum_{s=1}^{t-1} \hat{\ell}_s(i) \right) \quad i = 1, \ldots, N \)

- \( \hat{\ell}_t(i) = \begin{cases} \frac{\ell_t(i)}{P_t(\ell_t(i) \text{ observed})} & \text{if } \ell_t(i) \text{ is observed} \\ 0 & \text{otherwise} \end{cases} \)

Experts: \( \hat{\ell}_t = \ell_t \) (Hedge)

Bandits: Only one non-zero component in \( \hat{\ell}_t \) (Exp3)

Properties of importance weighting estimator

- \( \mathbb{E}_t [\hat{\ell}_t(i)] = \ell_t(i) \) unbiasedness

- \( \mathbb{E}_t [\hat{\ell}_t(i)^2] \leq \frac{1}{P_t(\ell_t(i) \text{ observed})} \) variance control
Regret bounds

\[ R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} P_t(I_t = i) \mathbb{E}_t \left[ \hat{\ell}_t(i)^2 \right] \right] \]

\[ \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(\ell_t(i) \text{ is observed})} \right] \]
Regret bounds

\[
R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} P_t(I_t = i) \mathbb{E}_t \left[ \hat{\ell}_t(i)^2 \right] \right]
\]

\[
\leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(\ell_t(i) \text{ is observed})} \right]
\]

- Experts: \( P_t(\ell_t(i) \text{ is observed}) = 1 \)
Regret bounds

\[ R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} \mathbb{P}_t(I_t = i) \mathbb{E}_t [\hat{\ell}_t(i)^2] \right] \]

\[ \leq \frac{\ln K}{\eta} + \frac{\eta}{2} T \sum_{t=1}^{T} \sum_{i=1}^{K} \mathbb{P}_t(I_t = i) \]

Experts: \( \mathbb{P}_t(\ell_t(i) \text{ is observed}) = 1 \)

implying \( R_T \leq \sqrt{T \ln K} \)
Regret bounds

$$R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} P_t(I_t = i) \mathbb{E}_t [\hat{\ell}_t(i)^2] \right]$$

$$\leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(\ell_t(i) \text{ is observed})} \right]$$

- **Experts:** $P_t(\ell_t(i) \text{ is observed}) = 1$
  implying $R_T \leq \sqrt{T \ln K}$

- **Bandits:** $P_t(\ell_t(i) \text{ is observed}) = P_t(I_t = i)$
Regret bounds

\[ R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} P_t(I_t = i) \mathbb{E}_t[\hat{\ell}_t(i)^2] \right] \]

\[ \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(I_t = i)} \]

**Experts:** \( P_t(\ell_t(i) \text{ is observed}) = 1 \)

implying \( R_T \leq \sqrt{T \ln K} \)

**Bandits:** \( P_t(\ell_t(i) \text{ is observed}) = P_t(I_t = i) \)

implying \( R_T \leq \sqrt{TK \ln K} \)
1. A brief digression
2. The silver bullet
3. The space tradeoff
4. The time tradeoff
Actions may return different amounts of feedback.
Actions may return different amounts of feedback.
Actions may return different amounts of feedback
Actions may return different amounts of feedback.
Active learning (revealing action)
Some old friends

Experts

Bandits
Regret bounds

\[ R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(\ell_t(i) \text{ is observed})} \right] \]
Regret bounds

\[
R_T \leq \frac{\ln K}{\eta} + \frac{\eta}{2} \mathbb{E} \left[ \sum_{t=1}^{T} \sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(\ell_t(i) \text{ is observed})} \right]
\]

Special case: symmetrical edges (undirected graph) with self loops

\[
\sum_{i=1}^{K} \frac{P_t(I_t = i)}{P_t(\ell_t(i) \text{ is observed})} = \sum_{i=1}^{K} \frac{P_t(I_t = i)}{\sum_{j : (i,j) \in E} P_t(I_t = j)} \leq \alpha_G \quad \text{independence number of } G
\]

Implying

\[
R_T \leq \sqrt{T\alpha_G \ln K}
\]

tight up to log factors
A vertex of $G$ is:

- **observable** if it has at least one incoming edge (possibly a self-loop)

1 and 4 are strongly observable
2 and 5 are weakly observable
3 is not observable
A vertex of $G$ is:

- **observable** if it has at least one incoming edge (possibly a self-loop)
- **strongly observable** if it has either a self-loop or incoming edges from all other vertices

1 and 4 are strongly observable
2 and 5 are weakly observable
3 is not observable
A characterization of feedback graphs

A vertex of $G$ is:

- **observable** if it has at least one incoming edge (possibly a self-loop)
- **strongly observable** if it has either a self-loop or incoming edges from all other vertices
- **weakly observable** if it is observable but not strongly observable
A characterization of feedback graphs

A vertex of G is:

- **observable** if it has at least one incoming edge (possibly a self-loop)
- **strongly observable** if it has either a self-loop or incoming edges from all other vertices
- **weakly observable** if it is observable but not strongly observable

1 and 4 are strongly observable
2 and 5 are weakly observable
3 is not observable

N. Cesa-Bianchi (UNIMI)
A vertex of $G$ is:

- **observable** if it has at least one incoming edge (possibly a self-loop)
- **strongly observable** if it has either a self-loop or incoming edges from all other vertices
- **weakly observable** if it is observable but not strongly observable

- 3 is not observable
- 2 and 5 are weakly observable
A characterization of feedback graphs

A vertex of $G$ is:

- **observable** if it has at least one incoming edge (possibly a self-loop)
- **strongly observable** if it has either a self-loop or incoming edges from all other vertices
- **weakly observable** if it is observable but not strongly observable

1 and 4 are strongly observable

2 and 5 are weakly observable

3 is not observable

- 3 is not observable
- 2 and 5 are weakly observable
- 1 and 4 are strongly observable
Minimax rates

\( G \) is strongly observable  \( R_T = \tilde{\Theta}(\sqrt{\alpha_G T}) \)  Experts, Bandits, Cops & Robbers

\( G \) is weakly observable  \( R_T = \tilde{\Theta}(T^{2/3}/\delta_G) \)

\( G \) is not observable  \( R_T = \Theta(T) \)  Hopeless game

\( \delta_G \) is the size of the smallest set that dominates all weakly observable nodes of \( G \)

The rates show that this setting is "partial-monitoring-complete"
Minimax rates

- **G is strongly observable**
  \[ R_T = \widetilde{\Theta}(\sqrt{\alpha_G T}) \]  Experts, Bandits
  Cops & Robbers

- **G is weakly observable**
  \[ R_T = \widetilde{\Theta}(T^{2/3} \delta_G) \]  Revealing Action

- \( \delta_G \) is the size of the smallest set that dominates all weakly observable nodes of \( G \)
Minimax rates

\[ R_T = \tilde{\Theta}\left(\sqrt{\alpha_G T}\right) \quad \text{Experts, Bandits, Cops & Robbers} \]

\[ R_T = \tilde{\Theta}\left(\frac{T^{2/3}}{\delta_G}\right) \quad \text{Revealing Action} \]

\[ R_T = \Theta(T) \quad \text{Hopeless game} \]

- \(\delta_G\) is the size of the smallest set that dominates all weakly observable nodes of \(G\)
Minimax rates

- **G is strongly observable**  \( R_T = \tilde{\Theta}(\sqrt{\alpha_G T}) \)  
  - Experts, Bandits
  - Cops & Robbers

- **G is weakly observable**  \( R_T = \tilde{\Theta}(T^{2/3} \delta_G) \)  
  - Revealing Action

- **G is not observable**  \( R_T = \Theta(T) \)  
  - Hopeless game

- \( \delta_G \) is the size of the smallest set that dominates all weakly observable nodes of **G**

- The rates show that this setting is "partial-monitoring-complete"
Some curious cases

Experts vs. Cops & Robbers

Presence of red loops does not affect minimax regret

$$R_T = \Theta(\sqrt{T \ln K})$$
Some curious cases

Experts vs. Cops & Robbers

Presence of red loops does not affect minimax regret $R_T = \Theta(\sqrt{T \ln K})$

Sharp transitions

With red loop: strongly observable with $\alpha(G) = K - 1$ $R_T = \tilde{\Theta}(\sqrt{KT})$

Without red loop: weakly observable with $\delta(G) = 1$ $R_T = \tilde{\Theta}(T^{2/3})$
Summary

1. A brief digression
2. The silver bullet
3. The space tradeoff
4. The time tradeoff
- $N$ agents sitting on the vertices of an unknown communication graph $G = (V, E)$
Bandit Networks

- $N$ agents sitting on the vertices of an unknown communication graph $G = (V, E)$
- Agents cooperate to solve a common bandit problem
N agents sitting on the vertices of an unknown communication graph $G = (V, E)$

Agents cooperate to solve a common bandit problem

Each agent runs an instance of the same algorithm
Communication protocol

Recall: Losses are the same for all agents

- All agents play *simultaneously* and exchange loss information across the network
- Information spreads in a non-instantaneous manner
- We study evolution of regret averaged over agents
The bandit protocol with fixed delay $d$

For each $t = 1, \ldots, T$ each agent $v \in V$ does the following:

1. Plays an action $I_t(v)$ drawn according to his private distribution $p_t(v)$ observing loss $\ell_t(I_t(v))$ (same loss vector for all agents)
For each $t = 1, \ldots, T$ each agent $v \in V$ does the following:

1. Plays an action $I_t(v)$ drawn according to his private distribution $p_t(v)$ observing loss $\ell_t(I_t(v))$ (same loss vector for all agents).

2. Sends to his neighbors the message

$$m_t(v) = \langle t, v, I_t(v), \ell_t(I_t(v)), p_t(v) \rangle$$
For each $t = 1, \ldots, T$ each agent $v \in V$ does the following:

1. Plays an action $I_t(v)$ drawn according to his private distribution $p_t(v)$ observing loss $\ell_t(I_t(v))$ (same loss vector for all agents)

2. Sends to his neighbors the message

$$m_t(v) = \langle t, v, I_t(v), \ell_t(I_t(v)), p_t(v) \rangle$$

3. Receives messages from his neighbors, forwarding those that are not older than $d$
The bandit protocol with fixed delay \( d \)

For each \( t = 1, \ldots, T \) each agent \( v \in V \) does the following:

1. Plays an action \( I_t(v) \) drawn according to his private distribution \( p_t(v) \) observing loss \( \ell_t(I_t(v)) \) (same loss vector for all agents)
2. Sends to his neighbors the message

\[
m_t(v) = \langle t, v, I_t(v), \ell_t(I_t(v)), p_t(v) \rangle
\]

3. Receives messages from his neighbors, forwarding those that are not older than \( d \)

- An agent receives a message from another agent with a delay equal to the shortest path between them
The bandit protocol with fixed delay $d$

For each $t = 1, \ldots, T$ each agent $v \in V$ does the following:

1. Plays an action $I_t(v)$ drawn according to his private distribution $p_t(v)$ observing loss $\ell_t(I_t(v))$ (same loss vector for all agents)
2. Sends to his neighbors the message
   $$m_t(v) = \langle t, v, I_t(v), \ell_t(I_t(v)), p_t(v) \rangle$$
3. Receives messages from his neighbors, forwarding those that are not older than $d$

- An agent receives a message from another agent with a delay equal to the shortest path between them
- A message sent by some agent $v$ at time $t$ will be received by all agents whose shortest-path distance from $v$ is at most $d$
Average welfare regret

\[ R_{T}^{\text{coop}} = \frac{1}{N} \sum_{v \in V} \mathbb{E} \left[ \sum_{t=1}^{T} \ell_{t}(I_{t}(v)) \right] - \min_{i=1, \ldots, k} \sum_{t=1}^{T} \ell_{t}(i) \]
Average welfare regret

\[ R_{T}^{\text{coop}} = \frac{1}{N} \sum_{v \in V} \mathbb{E} \left[ \sum_{t=1}^{T} \ell_t(I_t(v)) \right] - \min_{i=1,\ldots,K} \sum_{t=1}^{T} \ell_t(i) \]

Remarks

- Clearly, \( R_{T}^{\text{coop}} \leq \sqrt{TK \ln K} \) when agents run vanilla Exp3 (no cooperation)
Average welfare regret

\[ \mathbb{R}_T^{\text{coop}} = \frac{1}{N} \sum_{v \in V} \mathbb{E} \left[ \sum_{t=1}^T \ell_t(I_t(v)) \right] - \min_{i=1, \ldots, K} \sum_{t=1}^T \ell_t(i) \]

Remarks

- Clearly, \( \mathbb{R}_T^{\text{coop}} \leq \sqrt{TK \ln K} \) when agents run vanilla Exp3 (no cooperation)
- By using other agent’s plays, each agent may estimate \( \ell_t \) better (thus learning nearly at full info rate)
Average welfare regret

\[ R_{T}^{\text{coop}} = \frac{1}{N} \sum_{v \in V} \mathbb{E} \left[ \sum_{t=1}^{T} \ell_t(I_t(v)) \right] - \min_{i=1,\ldots,K} \sum_{t=1}^{T} \ell_t(i) \]

Remarks

- Clearly, \( R_{T}^{\text{coop}} \leq \sqrt{TK \ln K} \) when agents run vanilla Exp3 (no cooperation)
- By using other agent’s plays, each agent may estimate \( \ell_t \) better (thus learning nearly at full info rate)
- Delay \( d \) trades off between quality and quantity of information
Cooperative delayed loss estimator

Each agent $v$ uses the messages received from the other agents in order to estimate $\ell_t$ better

$$\hat{\ell}_t(i,v) = \begin{cases} 
\frac{\ell_{t-d}(i) \times B_{t-d}(i,v)}{P_{t-d}(B_{t-d}(i,v))} & \text{if } t > d \\
0 & \text{otherwise}
\end{cases}$$
Each agent $v$ uses the messages received from the other agents in order to estimate $\ell_t$ better

$$\hat{\ell}_t(i, v) = \begin{cases} \frac{\ell_{t-d}(i) \times B_{t-d}(i, v)}{\mathbb{P}_{t-d}(B_{t-d}(i, v))} & \text{if } t > d \\ 0 & \text{otherwise} \end{cases}$$

- $B_{t-d}(i, v)$ is the event: some agent in a $d$-neighborhood of $v$ played action $i$ at time $t - d$
Cooperative delayed loss estimator

Each agent $v$ uses the messages received from the other agents in order to estimate $\ell_t$ better

$$
\hat{\ell}_t(i, v) = \begin{cases} 
\frac{\ell_{t-d}(i) \times B_{t-d}(i, v)}{P_{t-d}(B_{t-d}(i, v))} & \text{if } t > d \\
0 & \text{otherwise}
\end{cases}
$$

- $B_{t-d}(i, v)$ is the event: some agent in a $d$-neighborhood of $v$ played action $i$ at time $t - d$
- Now $\hat{\ell}(v)$ may have many non-zero components (better estimate)
Cooperative delayed loss estimator

Each agent $v$ uses the messages received from the other agents in order to estimate $\ell_t$ better

$$
\hat{\ell}_t(i, v) = \begin{cases} 
\frac{\ell_{t-d}(i) \times B_{t-d}(i, v)}{P_{t-d}(B_{t-d}(i, v))} & \text{if } t > d \\
0 & \text{otherwise}
\end{cases}
$$

- $B_{t-d}(i, v)$ is the event: some agent in a $d$-neighborhood of $v$ played action $i$ at time $t - d$
- Now $\hat{\ell}(v)$ may have many non-zero components (better estimate)
- Agents need $p_{t-d}(v')$ in order to compute $P(B_{t-d}(i, v))$ Why?
Each agent $v$ uses the messages received from the other agents in order to estimate $\ell_t$ better

$$
\hat{\ell}_t(i, v) = \begin{cases} 
\frac{\ell_{t-d}(i) \times B_{t-d}(i, v)}{P_{t-d}(B_{t-d}(i, v))} & \text{if } t > d \\
0 & \text{otherwise}
\end{cases}
$$

- $B_{t-d}(i, v)$ is the event: some agent in a $d$-neighborhood of $v$ played action $i$ at time $t - d$
- Now $\hat{\ell}(v)$ may have many non-zero components (better estimate)
- Agents need $p_{t-d}(v')$ in order to compute $P(B_{t-d}(i, v))$ Why?
- A message $m_t(v')$ received by some agent $v$ is always used at time $t + d$ (even when $v' = v$)
Key inequality

\[ \mathbb{E} \left[ \sum_{i=1}^{K} \sum_{v \in V} \frac{P_t(I_t(v) = i)}{P_{t-d}(\ell_{t-d}(i) \text{ is observed by } v)} \right] \leq \frac{e}{1 + e^{-1}} (K \alpha_d + N) \]

\( \alpha_d \) is the independence number of the graph obtained from \( G \) by connecting any two vertices whose shortest path distance is at most \( d \).
Average welfare regret bound

\[
R_T^{\text{coop}} \sim \sqrt{\left( (d + 1) + \frac{K}{N} \alpha_d \right) \ln K}
\]

main term

unavoidable
Average welfare regret bound

\[ R^\text{coop}_T \sim \sqrt{\left( (d + 1) + \frac{K}{N} \alpha_d \right) \frac{T \ln K}{\text{main term}}} \]

Safe choice for delay

- \[ \alpha_d \leq \frac{2N}{d+2} \] for any connected graph
Average welfare regret bound

\[ R_{T}^{\text{coop}} \sim \sqrt{\left( (d + 1) + \frac{K}{N} \alpha_d \right) T \ln K} \]

Main term

Unavoidable

Safe choice for delay

- \( \alpha_d \leq \frac{2N}{d+2} \) for any connected graph
- Choose \( d = \sqrt{K} \)
Average welfare regret bound

\[ R_T^{\text{coop}} \overset{\sim}{=} \sqrt{\left( (d + 1) + \frac{K}{N} \alpha_d \right) \cdot T \ln K} \]

\[ \text{main term} \quad \text{unavoidable} \]

Safe choice for delay

- \( \alpha_d \leq \frac{2N}{d+2} \) for any connected graph
- Choose \( d = \sqrt{K} \)
- \( R_T^{\text{coop}} \overset{\sim}{=} \sqrt{K^{1/2}T \ln K} \)
Average welfare regret bound

\[ R_T^{\text{coop}} \sim \sqrt{\left( (d + 1) + \frac{K}{N} \alpha_d \right) T \ln K} \]

- **Main term**
- **Unavoidable**

**Safe choice for delay**

- \( \alpha_d \leq \frac{2N}{d+2} \) for any connected graph
- Choose \( d = \sqrt{K} \)
- \( R_T^{\text{coop}} \sim \sqrt{K^{1/2} T \ln K} \)
- This is better than \( \sqrt{KT} \) (minimax for non-cooperating bandits)
• Delays can be shorter in regions where the graph is dense
• Delays can be shorter in regions where the graph is dense
• This can be implemented using personalized \textit{time-to-live} parameters
- Delays can be shorter in regions where the graph is dense
- This can be implemented using personalized *time-to-live* parameters
- Regret improves
Remarks

- Delays can be shorter in regions where the graph is dense
- This can be implemented using personalized time-to-live parameters
- Regret improves

- Agents can use messages as they arrive (without waiting $d$ steps)
Remarks

- Delays can be shorter in regions where the graph is dense.
- This can be implemented using personalized \textit{time-to-live} parameters.
- Regret improves.

- Agents can use messages as they arrive (without waiting \(d\) steps).
- This implies that updates mix losses with different delays.
Remarks

- Delays can be shorter in regions where the graph is dense.
- This can be implemented using personalized time-to-live parameters.
- Regret improves.

- Agents can use messages as they arrive (without waiting $d$ steps).
- This implies that updates mix losses with different delays.
- We can put a prior over these delays and compute updates as averages.
Thanks for your attention!

Contributors

- Noga Alon
- Peter Auer
- Ofer Dekel
- Yoav Freund
- Claudio Gentile
- Tomer Koren
- Shie Mannor
- Yishay Mansour
- Rob Schapire
- Ohad Shamir