Introduction to bandit problems and algorithms

Sébastien Gerchinovitz

Cerfacs training October 2023

Outline

This lecture is a short introduction to bandit problems and algorithms.

For an in-depth treatment, we suggest the recent book *Bandit algorithms* by Lattimore and Szepesvári (2020). See also this tutorial or this blog.

Outline:

1. The K-armed bandit problem
2. Various extensions for numerous applications
3. An example in ad auction optimization
4. Next: Reinforcement Learning

1. The K-armed bandit problem
   - Setting
   - Well-known (but suboptimal) bandit algorithms
   - Better performances with refined algorithms

2. Various extensions for numerous applications
   - K-armed bandits: loosening the i.i.d. assumption
   - Bandit problems with more complex decision space
   - Best-arm identification

3. An example in ad auction optimization

4. Next: Reinforcement Learning

The Multi-Armed Bandit problem (MAB)

DEpendable & Explainable Learning

The Multi-Armed Bandit problem (MAB) is a toy problem that models sequential decision tasks where the learner must simultaneously exploit their knowledge and explore unknown actions to gain knowledge for the future (exploration-exploitation tradeoff).

Toy example: playing in a casino.

- Imagine we are given 1000 USD that we can use on 10 different slot machines (or *one-armed bandits*), 1 USD each.
- The average reward may vary from one slot machine to another. We initially do not know which machine is optimal.
- What is the best strategy to optimize our cumulative reward after 1000 rounds?
- We should both try all machines (exploration) while playing an empirically good machine sufficiently often (exploitation).

A more serious application

Imagine you are a doctor:

- Patients visit you *one after another* for a given disease.
- You prescribe one of the (say) *5 treatments* available.
- The treatments are *not equally efficient*.
- You do not know which one is the best, you *observe the effect* of the prescribed treatment on each patient

$\rightsquigarrow$ What should you do?

- You must choose each prescription using only the *previous observations*.
- Your goal is not to estimate each treatment's efficiency precisely, but to *heal as many patients as possible* ($\neq$ clinical trials).

Formal statement of the MAB problem

We write $g_t(i)$ for the reward (gain) of arm $i \in \{1, \ldots, K\}$ at round $t \geqslant 1$. We assume that the sequence of reward vectors $g_1, g_2, \ldots \in \mathbb{R}^K$ is chosen at the beginning of the game, and is i.i.d. for the moment. We set:

$$\mu_i := \mathbb{E}\big[g_1(i)\big] \qquad \text{and} \qquad \mu^\star := \max_{1 \leqslant i \leqslant K} \mu_i \,.$$

Online protocol: at each round $t \in \mathbb{N}^*$,

1. The learner chooses an action $I_t \in \{1, \ldots, K\}$, possibly at random.
2. The learner receives and observes the reward $g_t(I_t)$, but does not observe the reward $g_t(i)$ they would have got had they played another action $i \neq I_t$.

Goal: minimize the (pseudo) regret

$$R_T := \max_{1 \leqslant i \leqslant K} \mathbb{E}\left[\sum_{t=1}^{T} g_t(i)\right] - \mathbb{E}\left[\sum_{t=1}^{T} g_t(I_t)\right] = T\mu^\star - \mathbb{E}\left[\sum_{t=1}^{T} g_t(I_t)\right] .$$

A low regret means that the learner played (in expectation) almost as good as the best action, which is unknown to the learner.

The Explore-Then-Commit algorithm

DEEL
DEpendable & Explainable Learning

Explore-Then-Commit (ETC)

Parameter: number $m \in \mathbb{N}^*$ of initial draws for each arm.

1. At each round $t \in \{1, \ldots, mK\}$, choose action $I_t = (t \bmod K) + 1$.
2. At each round $t \geqslant mK + 1$, choose the action that was empirically best after the first phase: $I_t = \mathrm{argmax}_{1 \leqslant i \leqslant K} \, \widehat{\mu}_i(mK)$.

Theoretical guarantee: if the reward vectors $g_1, g_2, \ldots \in \mathbb{R}^K$ are i.i.d. and each $g_1(i) - \mu_i$ is subgaussian with variance factor σ^2, then ETC satisfies (see, e.g., Thm 6.1 by Lattimore and Szepesvári 2020)

$$R_T \leqslant m \sum_{i=1}^{K} \Delta_i + T \sum_{i=1}^{K} \Delta_i \exp\left(-\frac{m\Delta_i^2}{4\sigma^2}\right) ,$$

where $\Delta_i = \mu^\star - \mu_i$ is the suboptimality gap of arm i.

Consequence: for $K = 2$ arms with gap $\Delta > 0$, tuning $m \approx \log(T\Delta^2)/\Delta^2$ yields $R_T \lesssim \log(T\Delta^2)/\Delta$.

The Explore-Then-Commit algorithm

Explore-Then-Commit (ETC)

Parameter: number $m \in \mathbb{N}^*$ of initial draws for each arm.

1. At each round $t \in \{1, \dots, mK\}$, choose action $I_t = (t \bmod K) + 1$.
2. At each round $t \geqslant mK + 1$, choose the action that was empirically best after the first phase: $I_t = \operatorname{argmax}_{1 \leqslant i \leqslant K} \widehat{\mu}_i(mK)$.

Consequence: for $K = 2$ arms with gap $\Delta > 0$, tuning $m \approx \log(T\Delta^2)/\Delta^2$ yields $R_T \lesssim \log(T\Delta^2)/\Delta$.

- Issue 1: if T is unknown, the choice of m is impractical. Besides, the regret R_T usually grows linearly with T if m is fixed and $T \to +\infty$. Completely stopping exploring if we do not know T is a bad idea!
- Issue 2: Δ is usually unknown.

DEEL
DEpendable & Explainable Learning

Definition

Let $v \in \mathbb{R}_+$. A real random variable X is said to be subgaussian with variance factor v iff

$$\forall \lambda \in \mathbb{R}\,, \qquad \mathbb{E}\left[e^{\lambda X}\right] \leqslant \exp\left(\frac{\lambda^2 v}{2}\right)\,. \tag{1}$$

It can be shown that a subgaussian r.v. has finite moments at all orders, and has mean 0 and variance at most v.

Examples:

- if $X \sim \mathcal{N}(\mu, \sigma^2)$, then $X - \mu$ satisfies (1) with equality for $v = \sigma^2$;
- if $X \in [a, b]$ is a bounded random variable, then $X - \mathbb{E}[X]$ satisfies (1) with $v = (b-a)^2/4$.

Let $v > 0$. If X is subgaussian with variance factor v, then by Markov's inequality, for all $x > 0$ and all $\lambda > 0$,

$$\mathbb{P}(X \geqslant x) = \mathbb{P}\left(e^{\lambda X} > e^{\lambda x}\right) \leqslant e^{-\lambda x}\,\mathbb{E}\left[e^{\lambda X}\right] \leqslant e^{-\lambda x + \lambda^2 v/2} .$$

Optimizing in λ yields $\mathbb{P}(X \geqslant x) \leqslant e^{-x^2/(2v)}$ for all $x > 0$, and $\mathbb{P}(X \leqslant -x) \leqslant e^{-x^2/(2v)}$ as well. For n independent r.v., we have:

Lemma (Subgaussian deviation inequality for the empirical mean)

Let $X_1, X_2, \ldots$ be i.i.d. real random variables such that $X_1 - \mu$ is subgaussian with variance factor σ^2. Then, the empirical mean $\widehat{\mu}_n = \frac{1}{n}\sum_{k=1}^{n} X_k$ satisfies, for all $n \in \mathbb{N}^*$ and $x > 0$,

$$\mathbb{P}\left(\widehat{\mu}_n \geqslant \mu + x\right) \leqslant e^{-nx^2/(2\sigma^2)}$$

$$\mathbb{P}\left(\widehat{\mu}_n \leqslant \mu - x\right) \leqslant e^{-nx^2/(2\sigma^2)}$$

The deviation probability bounds decrease exponentially fast with n and x^2, but increase with σ^2.

The ε-Greedy algorithm

DEEL
DEpendable & Explainable Learning

ε-Greedy

Parameters: $\varepsilon_1, \varepsilon_2, \ldots \in (0, 1]$.
At each round $t \geqslant 1$,

1. let J_t be the best arm so far (highest empirical average);
2. play J_t with probability $1 - \varepsilon_t$ or a random uniform arm with probability ε_t.

Theoretical guarantee: Auer et al. (2002a) proved that if the reward vectors $g_1, g_2, \ldots \in [0, 1]^K$ are i.i.d. and if $\varepsilon_t \approx K/(\Delta^2 t)$, then ε-Greedy satisfies

$$R_T \lesssim \frac{K \log T}{\Delta^2},$$

where the gap Δ is the difference between the reward expectations of the best arm and the next best arm.

Now, T is not required to tune the algorithm, but Δ still is.

This algorithm follows the 'Optimism in face of uncertainty' principle.

UCB1 (Upper Confidence Bound) `UCB.avi`

Initialization: play each arm once.

At each round $t \geqslant K+1$,

1. play arm $I_t \in \mathrm{argmax}_{1 \leqslant i \leqslant K} \left\{ \widehat{\mu}_{t-1}(i) + \sqrt{\dfrac{2\log t}{T_i(t-1)}} \right\}$, where

   $\widehat{\mu}_{t-1}(i)$ is the average reward of arm i up to round $t-1$, and
   $T_i(t-1)$ is the number of times arm i was played.

Theoretical guarantee: Auer et al. (2002a) proved that if the reward vectors $g_1, g_2, \ldots \in [0,1]^K$ are i.i.d., then UCB1 satisfies

$$R_T \leqslant \sum_{i:\Delta_i>0} \frac{8\log T}{\Delta_i} + \left(1+\frac{\pi^2}{3}\right)\sum_{i=1}^{K}\Delta_i\,,$$

where $\Delta_i = \mu^\star - \mu_i$ is the difference between the reward expectations of the best arm and the i-th arm. (Now, the algorithm does not use the Δ_i.)

Better performances with refined algorithms

Warning: UCB should not be used in practice!

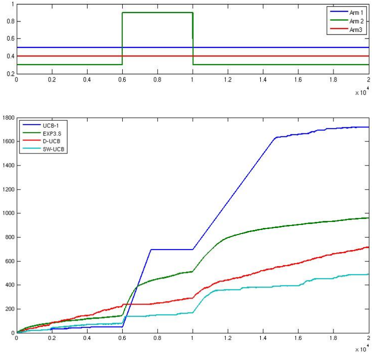
UCB uses unnecessarily large confidence intervals (based on Hoeffding's inequality), which results in a suboptimal multiplicative constant before $\log(T)$, and slow convergence in practice.

Refined algorithms with better (and sometimes optimal) performances include:

- algorithms using tighter confidence intervals (e.g., KL-UCB or KL-UCB-switch);
- Bayesian algorithms (e.g., Thompson sampling).

1. The K-armed bandit problem
   - Setting
   - Well-known (but suboptimal) bandit algorithms
   - Better performances with refined algorithms
2. Various extensions for numerous applications
   - K-armed bandits: loosening the i.i.d. assumption
   - Bandit problems with more complex decision space
   - Best-arm identification
3. An example in ad auction optimization
4. Next: Reinforcement Learning

- Changepoint: reward distributions change *abruptly*
- Goal: *follow the best arm*
- Application: scanning tunnelling microscope



- Variants D-UCB et SW-UCB including a progressive *discount* of the past
- Bounds $O(\sqrt{n \log n})$ are proved, which is (almost) optimal

DEEL
DEpendable & Explainable Learning

We now consider arbitrary reward vectors $g_1, g_2, \ldots \in [0,1]^K$ (not necessarily drawn i.i.d. from a given distribution).

Exp3 algorithm

Parameters: $\eta_1, \eta_2, \ldots > 0$.

At each round $t \geqslant 1$,

1. compute the weight vector $w_t = (w_t(1), \ldots, w_t(K))$ given by

$$w_t(i) = \frac{\exp\Big(\eta_t \sum_{s=1}^{t-1} \tilde{g}_s(i)\Big)}{\sum_{j=1}^{K} \exp\Big(\eta_t \sum_{s=1}^{t-1} \tilde{g}_s(j)\Big)} \,, \quad 1 \leqslant i \leqslant K \;;$$

   where $\tilde{g}_s(i) = 1 - \frac{1-g_s(i)}{w_s(i)} \mathbb{1}_{I_s = i}$ is an unbiased estimator of $g_s(i)$;

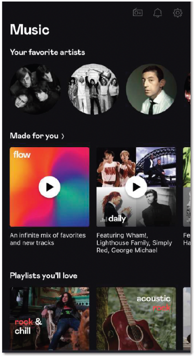
2. draw I_t at random such that $\mathbb{P}(I_t = i) = w_t(i)$.

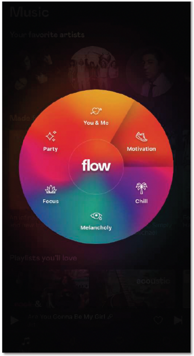
Theoretical guarantee: Auer et al. (2002b) proved $R_T \leqslant 2\sqrt{T\,K \ln K}$ with $\eta_t = \sqrt{\ln(K)/(tK)}$, for arbitrary reward vectors $g_1, g_2, \ldots \in [0,1]^K$. (Worst guarantee than UCB1, but more robust.)

Combinatorial bandits (1)

Sequentially choose an (ordered) subset of arms from a huge set.

Example: a recommender system suggests a list of products from a huge catalogue to any new customer on the website, with the goal of maximizing the total number of clicks or the total revenue.

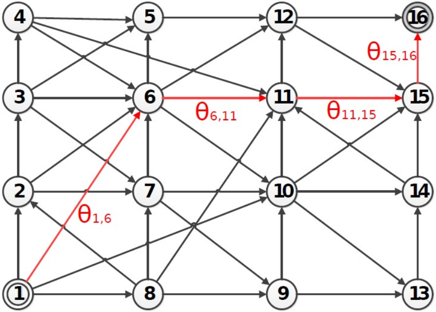




Source: `https://www.deezer.com/`

DEEL
DEpendable & Explainable Learning

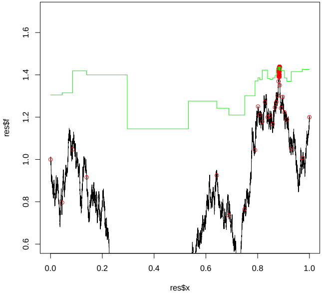
- Sequentially choose a path in a graph (with costs on edges).



Source: path routing example of Combes and Proutière in
`https://www.sigmetrics.org/sigmetrics2015/tutorial_sigmetrics.pdf`

- Sequentially choose a perfect matching in a complete bipartite graph (assignment problem).

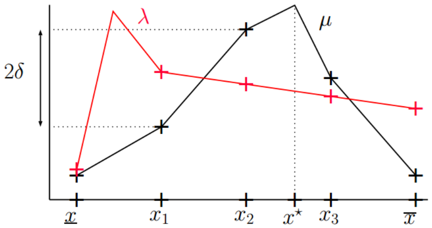
- Goal: sequentially play almost as good as the maximum of a function $f : C \subset \mathbb{R}^d \to \mathbb{R}$ that we observe (possibly) with noise.



- Various possible models : f has a certain regularity (e.g., Lipschitz or gradient-Lipschitz), f is the realization of a Gaussian Process, etc.
- Several algorithms: zooming algorithm, HOO, GP-UCB, etc (and other algorithms for the simple regret).

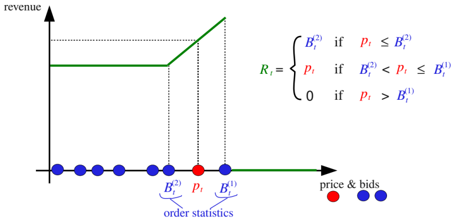
Unimodal bandits without smoothness: trisection algorithms, and better (Combes and Proutiere, 2014).

Application to internet network traffic optimization.



Reserve Price Optimization in Second-price Auctions (Cesa-Bianchi et al., 2015).

Application to ad placement.



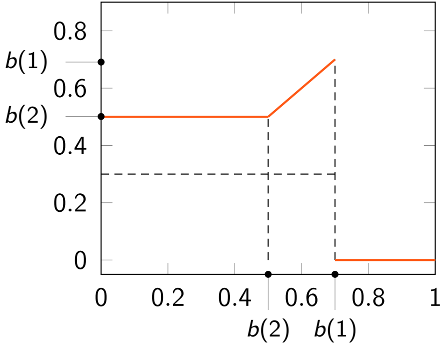
Ad auction:

- Online advertising: consider a publisher (seller) who want to sell an ad space to advertisers (buyers) through second-price auctions managed by an ad exchange.
- For each impression (ad display) created on the publisher's website, the ad exchange runs an auction on the fly.

Second-price auction:

- Simultaneously, all buyers propose a price (bid) to the ad exchange.
- The buyer with the highest bid wins the auction but pays the second highest price.
- This is a truthful mechanism.

- The seller has an additional degree of freedom: the reserve price, which corresponds to the minimal revenue they are willing to get.
- Before the auction, the seller communicates a reserve price y to the ad exchange (the reserve price is unknown to the buyers).
- If the reserve price y is larger than the highest bid $b(1)$, the auction is lost. Otherwise, the buyer with the highest bid wins the auction.
- The winner pays the maximum of the second-highest bid $b(2)$ and the reserve price y. The seller's revenue is $g(y) = \max\{b(2), y\}\mathbb{1}_{b(1) \geqslant y}$.



Example: online reserve price optimization (3)

DEEL
DEpendable & Explainable Learning

Assume now that the publisher participates to a series of auctions. The task of sequentially optimizing the reserve price can be phrased as a continuum-armed bandit problem: at each round $t \geqslant 1$,

- the seller sets a reserve price $\widehat{y}_t \in [0,1]$;
- simultaneously, a set of buyers propose bids $b_t(1) \geqslant b_t(2) \geqslant \cdots \in [0,1]$ (sorted in decreasing order);
- the seller receives and observes the revenue $g_t(\widehat{y}_t) = \max\{b_t(2), \widehat{y}_t\}\mathbb{1}_{b_t(1) \geqslant \widehat{y}_t}$.

Cesa-Bianchi et al. (2015) proposed an algorithm for the case when the bids are i.i.d. accross the buyers and time. They proved a $\tilde{\mathcal{O}}(\sqrt{T})$ upper bound on the (pseudo) regret

$$R_T := \sup_{y \in [0,1]} \mathbb{E}\left[\sum_{t=1}^{T} g_t(y)\right] - \mathbb{E}\left[\sum_{t=1}^{T} g_t(\widehat{y}_t)\right] .$$

Before choosing the arm $I_t \in \{1, \ldots, K\}$ or (more generally) the action $\widehat{y}_t \in \mathcal{Y}$, the learner has access to a context $x_t \in \mathcal{X}$.

Example: in ad auctions, the context may contain different properties of the customer or of the ad space.

General setting = contextual bandits: at each round $t \in \mathbb{N}^*$,

1. The environment reveals a context $x_t \in \mathcal{X}$.
2. The learner chooses an action $\widehat{y}_t \in \mathcal{Y}$, possibly at random.
3. The learner receives and observes a reward $g_t(\widehat{y}_t)$.

The goal is now to minimize the pseudo regret w.r.t. a (nonparametric) set of functions $\mathcal{F} \subset \mathcal{Y}^{\mathcal{X}}$ (e.g., Cesa-Bianchi et al. 2017):

$$R_T := \sup_{f \in \mathcal{F}} \mathbb{E}\left[\sum_{t=1}^{T} g_t\big(f(x_t)\big)\right] - \mathbb{E}\left[\sum_{t=1}^{T} g_t(\widehat{y}_t)\right] .$$

Best-arm identification

Also sometimes called pure exploration.

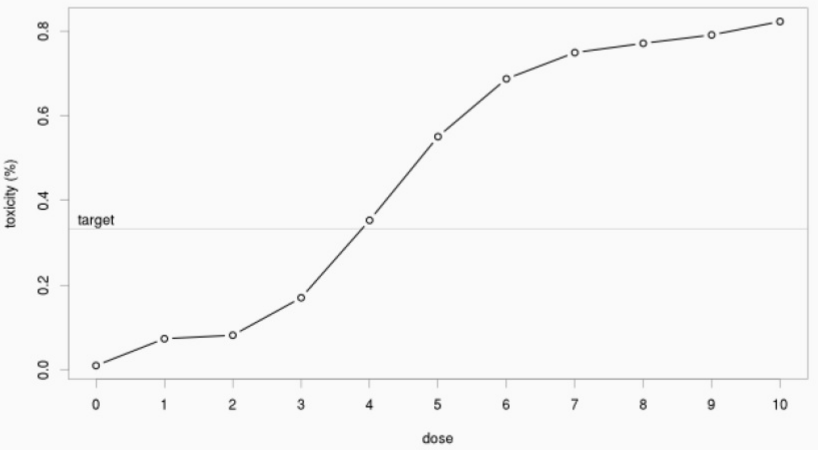
- Previous goal: maximize the cumulative reward.
- Now: identify arm with maximal expectation: $i^* \in \operatorname{argmax}_{1 \leqslant i \leqslant K} \mu_i$.
  For example, given δ, minimize the expected number of trials $\mathbb{E}[\tau_\delta]$ while ensuring the final recommandation $\widehat{i}$ is most probably correct:

$$\mathbb{P}\big(\widehat{i} \neq i^*\big) \leqslant \delta \,.$$

Applications:

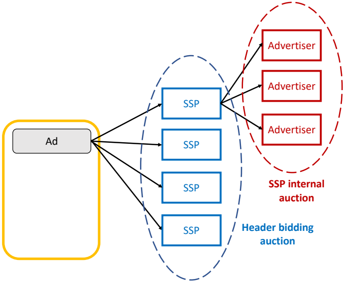
- clinical trials
- A/B testing (for, e.g., website design)
- continuous action space: zero-order stochastic optimization

See, e.g., Garivier and Kaufmann (2016).


toxicity (%)
0.0
0.2
0.4
0.6
0.8
target
0
1
2
3
4
5
6
7
8
9
10
dose

And much more!

An example of a continuum-armed bandit problem in ad auction optimization.
Joint work: Jauvion et al. (2018).



- A supply-side plateform (SSP) runs repeated second-price auctions where several advertizers compete to display ads on a publisher's ad space.
- After running any second-price auction, the SSP wants to minimize the price they offer to the publisher (the SSP gains the difference).
- However, the publisher might not accept this price since they are asking to several SSPs simultaneously for a given price (first-price auction).

More details on `https://alephd.github.io/assets/header_bidding/slides/`.

Take-home message: bandits = exploration-exploitation tradeoff.

Bandit problems are sequential decision models where the learner must simultaneously:

- exploit their current knowledge;
- explore unknown actions to gain knowledge for the future.

Not thinking about the future can be terribly bad!

There are multiple variants of the simple K-armed bandit problem that have been designed for numerous applications.

There are also pure-exploration bandit problems.

1. The K-armed bandit problem
   - Setting
   - Well-known (but suboptimal) bandit algorithms
   - Better performances with refined algorithms
2. Various extensions for numerous applications
   - K-armed bandits: loosening the i.i.d. assumption
   - Bandit problems with more complex decision space
   - Best-arm identification
3. An example in ad auction optimization
4. **Next: Reinforcement Learning**

Next: MDP and Reinforcement Learning

DEpendable & Explainable Learning

Bandits

- Bandit models are simple models that stress the importance to combine exploitation with exploration.
- Yet, making an action does not change the state of the environment.

Reinforcement Learning

- RL studies "learning from interaction to achieve a goal".
- Markov Decision Processes are more general models that include a state that can evolve over time, based on the actions of the learner.
- Example: inverted pendulum `https://www.youtube.com/watch?v=Lt-KLtkDlh8`
- See the book by Sutton and Barto (2018),
  and Erwan Le Pennec's reading notes:
  `http://www.cmap.polytechnique.fr/~lepennec/files/RL/Sutton.pdf`

References I

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.*, 47:235–256, 2002a.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002b.

Nicolò Cesa-Bianchi, Pierre Gaillard, Claudio Gentile, and Sébastien Gerchinovitz. Algorithmic chaining and the role of partial feedback in online nonparametric learning. In *Proceedings of the 2017 Conference on Learning Theory*, volume 65, pages 465–481, 2017.

NicolÃ² Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Regret minimization for reserve prices in second-price auctions. *IEEE Transactions on Information Theory*, 61 (1):549–564, 2015.

Richard Combes and Alexandre Proutiere. Unimodal bandits: Regret lower bounds and optimal algorithms. In *Proceedings of ICML 2014*, 2014.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 998–1027, 2016.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for non-stationary bandit problems. In *Proceedings of ALT 2011*, pages 174–188, 2011.

Grégoire Jauvion, Nicolas Grislain, Pascal Dkengne Sielenou, Aurélien Garivier, and Sébastien Gerchinovitz. Optimization of a SSP's header bidding strategy using Thompson Sampling. In *Proceedings of SIGKDD 2018, Applied Data Science track*, 2018.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020. URL http://downloads.tor-lattimore.com/banditbook/book.pdf.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning*. 2018. URL http://incompleteideas.net/book/the-book.html.