Decomposition-Coordination Method for Finite Horizon Bandit Problems

Michel de Lara, Benjamin Heymann, Jean-Philippe Chancelier

Decomposition-Coordination Method for Finite Horizon Bandit Problems

Michel De Lara*, Benjamin Heymann†, Jean-Philippe Chancelier*

June 1, 2021

Abstract

Optimally solving a multi-armed bandit problem suffers the curse of dimensionality. Indeed, resorting to dynamic programming leads to an exponential growth of computing time, as the number of arms and the horizon increase. We introduce a decomposition-coordination heuristic, DeCo, that turns the initial problem into parallelly coordinated one-armed bandit problems. As a consequence, we obtain a computing time which is essentially linear in the number of arms. In addition, the decomposition provides a theoretical lower bound on the regret. For the two-armed bandit case, dynamic programming provides the exact solution, which is almost matched by the DeCo heuristic. Moreover, in numerical simulations with up to 100 rounds and 20 arms, DeCo outperforms classic algorithms (Thompson sampling and Kullback-Leibler upper-confidence bound) and almost matches the theoretical lower bound on the regret for 20 arms.

Keywords. multi-armed bandit problem, dynamic programming, decomposition-coordination, DeCo heuristic

1 Introduction

A multi-armed bandit (MAB) is a mathematical model for sequential decision making under partial feedback. At each round, a decision maker selects an arm, and then the arm yields a random reward that depends on the intrinsic characteristics of the arm, which are unknown to the decision maker. The selection of an arm hence serves two purposes: amassing reward and acquiring information on the arm, to be used in the future rounds. For this reason, MABs embody the well-known exploration-exploitation tradeoff and have concentrated the attention of several research communities (reinforcement learning, statistics, operations research, economics...).

*CERMICS, Ecole des Ponts, Marne-la-Vallée, France
†Criteo, Paris, France

The first occurrence of MABs in the literature was motivated by clinical trials [20], but the rise of the digital economy has unlocked many new applications [8, 17, 21]. As observed in [6], two schools emerged from the early works on MABs. The first school follows [1] and aims at maximizing the expected total reward over a discounted horizon, and envisions the multi-armed bandit as a Markov Decision Process (MDP). The pioneering breakthrough is the Gittins Index Theorem [12] which provides a way to decompose the problem into tractable sub-problems, one per arm. The second school follows Robbins formulation [18] and minimizes an expected regret over a finite horizon. The seminal work [15] identifies an asymptotically efficient policy. Other problem formulations and approaches were proposed following this milestone [16, 3].

In this article, we take the MDP perspective, and aim at minimizing the Bayesian regret of a binary bandit over a finite horizon. Theoretically, such problem could be addressed with dynamic programming, but this is not feasible because of the curse of dimensionality: the problem size grows exponentially in the number of arms. We leverage the ideas from [4] to show how time decomposition (dynamic programming) can be made compatible with arm decomposition. Indeed, it is illustrated how a structured, large scale intertemporal maximization problem can be transformed into a collection of parameterized, simpler, intertemporal subproblems by relaxing coupling constraints. Thus doing, one obtains a collection of local value functions, one per subproblems, all functions of a common coordinating parameter process. After optimizing this latter, one sums the local value functions and uses the resulting surrogate global value function in the online phase of the Bellman equation. This gives both a theoretical upper bound for the original maximization problem, and a heuristic online policy. Our contribution is twofold: first, we introduce a novel way to establish a lower bound for the optimal regret; second, we derive, from this principled approach, a policy that achieves state of the art performances on the experiments we ran.

The paper is organized as follows. In Sect. 2, we describe the stochastic multi-armed bandit problem, show how it can be framed in the multistage stochastic optimal control formalism, and then adapt the method in [4] to finally show how this control problem can be treated by decomposition-coordination. In Sect. 3, we benchmark DeCo against Thomson Sampling (Ts) [20, 8], Kullback-Leibler upper-confidence bound (Kl-Ucb) [10] and, in the case of two arms, the exact dynamic programming resolution.

2 Decomposition-coordination method for the bandit problem

In §2.1, we describe the stochastic multi-armed bandit problem, and show how it can be framed in the multistage stochastic optimal control formalism. In §2.2, we adapt the method in [4], and we show how this control problem can be treated by decomposition-coordination.

2.1 Multistage stochastic optimal control formulation

For any integers $a \leq b$, $[\![a,b]\!]$ denotes the subset $\{a, a+1, \ldots, b-1, b\}$. We consider a decision-maker (DM) who selects an arm a in a finite set A, at each discrete time step t in the set $[\![0, T-1]\!]$, where $T \geq 1$ is an integer. Thus doing, the arm a delivers, at the end of the time interval $[t, t+1[$, a random variable[1] $\mathbf{W}_{t+1}^a$ that takes two values[2] in the set $\{\mathtt{B}, \mathtt{G}\}$ ("bad" $\mathtt{B}$, "good" $\mathtt{G}$) and with unknown parameter $\bar{p}_{\mathtt{G}}^a$, the probability of the event $\mathbf{W}_{t+1}^a = \mathtt{G}$. The parameter $\bar{p}_{\mathtt{G}}^a \in [0,1]$ is unknown to the DM, which we formalize below.

Probabilistic model Let $\Sigma = \big\{p = (p^{\mathtt{B}}, p^{\mathtt{G}}) \in \mathbb{R}^2 \,\big|\, p^{\mathtt{B}} \geq 0\ ,\ \ p^{\mathtt{G}} \geq 0\ ,\ \ p^{\mathtt{B}} + p^{\mathtt{G}} = 1\big\}$ be the one-dimensional simplex[3]. For any $p = (p^{\mathtt{B}}, p^{\mathtt{G}}) \in \Sigma$, we denote by $\mathcal{B}(p^{\mathtt{B}}, p^{\mathtt{G}}) = \bigotimes_{t=1}^{T} \big(p^{\mathtt{B}}\delta_{\mathtt{B}} + p^{\mathtt{G}}\delta_{\mathtt{G}}\big)$ the probability on $\{\mathtt{B}, \mathtt{G}\}^T$ which corresponds to a sequence of independent (Bernoulli) random variables with values in $\{\mathtt{B}, \mathtt{G}\}$. For any $\{p^a\}_{a\in A} = \{(p^{\mathtt{B}a}, p^{\mathtt{G}a})\}_{a\in A} \in \prod_{a\in A}\Sigma$, we consider the probability $\bigotimes_{a\in A}\mathcal{B}(p^{\mathtt{B}a}, p^{\mathtt{G}a})$ on the product space $\prod_{a\in A}\{\mathtt{B}, \mathtt{G}\}^T$, which corresponds to independence between arms in A. We denote by $\mathbb{E}_{\{p^a\}_{a\in A}}$ the corresponding mathematical expectation. We suppose that the DM holds a prior π_0^a over the unknown $p^a = (p^{\mathtt{B}a}, p^{\mathtt{G}a}) \in \Sigma$, for every arm $a \in A$. In practice, we will consider a beta distribution $\beta(n^{\mathtt{B}}, n^{\mathtt{G}})$ on Σ, with positive integers $n^{\mathtt{B}} > 0$ and $n^{\mathtt{G}} > 0$ as parameters.

We consider the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ where $\Omega = \prod_{a\in A}\Sigma \times \{\mathtt{B}, \mathtt{G}\}^T$, $\mathcal{F} = 2^\Omega$, $\mathbb{P} = \bigotimes_{a\in A}\pi_0^a\big(\mathrm{d}(p^{\mathtt{B}a}, p^{\mathtt{G}a})\big) \otimes \mathcal{B}(p^{\mathtt{B}a}, p^{\mathtt{G}a})$. Then, $\mathbf{W}^a = \{\mathbf{W}_t^a\}_{t\in[\![1,T]\!]}$ denotes the coordinate mappings for every arm $a \in A$, with $\mathbf{W}_t^a$ a random variable having values in the set $\{\mathtt{B}, \mathtt{G}\}$. For a given family $\{(\bar{p}_{\mathtt{B}}^a, \bar{p}_{\mathtt{G}}^a)\}_{a\in A} \in \prod_{a\in A}\Sigma$ and for $\pi_0^a = \delta_{(\bar{p}_{\mathtt{B}}^a, \bar{p}_{\mathtt{G}}^a)}$, for every arm $a \in A$, the family $\{\mathbf{W}_t^a\}_{a\in A, t\in[\![1,T]\!]}$ consists of independent random variables, where $\mathbf{W}_t^a$ has (Bernouilli) probability distribution with parameter $\bar{p}_{\mathtt{G}}^a \in [0,1]$, that is, $\mathbb{P}\big(\mathbf{W}_t^a = \mathtt{B}\big) = 1 - \bar{p}_{\mathtt{G}}^a$ and $\mathbb{P}\big(\mathbf{W}_t^a = \mathtt{G}\big) = \bar{p}_{\mathtt{G}}^a$. With this probabilistic model, we represent the sequential independent outcomes of $|A|$ independent arms.

Decision model We consider a sequence $\mathbf{U} = \{\mathbf{U}_t\}_{t\in[\![0,T-1]\!]}$ of random variables (on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$), where $\mathbf{U}_t = \{\mathbf{U}_t^a\}_{a\in A}$, $\mathbf{U}_t^a \in \{0,1\}$, $\forall a \in A$, $\forall t \in [\![0, T-1]\!]$. Their possible values in $\{0,1\}$, represent that either arm a has been selected at the beginning of the time interval $[t, t+1[$ ($\mathbf{U}_t^a = 1$) or not ($\mathbf{U}_t^a = 0$). Since, at each given time, one and only one arm has to be selected, we add the constraint

$$\sum_{a\in A} \mathbf{U}_t^a = 1\ ,\ \ \forall t \in [\![0, T-1]\!]\ . \tag{1}$$

This way of modeling the selection of a unique arm is not the most common in the bandit literature, but we can find it for example in [6].

[1] The shifted index $t+1$ is here to indicate that the random variable $\mathbf{W}_{t+1}^a$ materializes during the time interval $[t, t+1[$.

[2] We call these two values "bad" (for $\mathtt{B}$), and "good" (for $\mathtt{G}$), and not $\{0,1\}$ to avoid confusion with the possible values for the controls ("do not select arm", "select arm"). In fact, we take two values for the sake of simplicity, but we could have taken a finite or even infinite number of values.

[3] For the sake of symmetry between outcomes $\mathtt{B}$ and $\mathtt{G}$, we do not identify the simplex Σ with the unit segment $[0,1]$ by the mapping $\Sigma \ni (p^{\mathtt{B}}, p^{\mathtt{G}}) \mapsto p^{\mathtt{B}} \in [0,1]$.

Information and admissible controls When the arm a has been selected at stage t (that is, when $\mathbf{U}_t^a = 1$), the DM observes the outcome, in the set $\{\mathtt{B}, \mathtt{G}\}$, of the random variable $\mathbf{W}_{t+1}^a$. When the arm a has not been selected at stage t (that is, when $\mathbf{U}_t^a = 0$), the DM observes nothing. Thus, the DM observes the random variable $\mathbf{Y}_{t+1} = \left\{\mathbf{U}_t^a \mathbf{W}_{t+1}^a\right\}_{a \in A}$, $\forall t \in [\![0, T-1]\!]$, and the admissible controls $\mathbf{U} = \{\mathbf{U}_t\}_{t \in [\![0,T-1]\!]}$ are those that satisfy

$$\sigma(\mathbf{U}_t) \subset \sigma(\mathbf{Y}_0, \mathbf{U}_0, \mathbf{Y}_1, \ldots, \mathbf{U}_{t-1}, \mathbf{Y}_t) \ , \ \ \forall t \in [\![0, T-1]\!] \ , \tag{2}$$

where $\sigma(\mathbf{Z}) \subset \mathcal{F}$ is the σ-field generated by the random variable $\mathbf{Z}$ on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$.

Random rewards We suppose given a family $\{L_t^a\}_{a \in A, t \in [\![0,T-1]\!]}$ of functions $L_t^a : \{\mathtt{B}, \mathtt{G}\} \to \mathbb{R}$, that represent instantaneous rewards as follows. When the arm a has been selected at stage t (that is, when $\mathbf{U}_t^a = 1$), the random variable $\mathbf{W}_{t+1}^a$ materializes and the DM receives the payoff $1 \times L_t^a(\mathbf{W}_{t+1}^a) = \mathbf{U}_t^a L_t^a(\mathbf{W}_{t+1}^a)$. When the arm a has not been selected at stage t (that is, when $\mathbf{U}_t^a = 0$), the DM receives the payoff $0 = \mathbf{U}_t^a L_t^a(\mathbf{W}_{t+1}^a)$. Thus, the total random reward associated with the control $\mathbf{U} = \{\mathbf{U}_t\}_{t \in [\![0,T-1]\!]}$ is given by $\sum_{t=0}^{T-1} \sum_{a \in A} \mathbf{U}_t^a L_t^a(\mathbf{W}_{t+1}^a)$.

Optimality criteria in the Bayesian framework Let $\Delta(\Sigma)$ denote the set of probability distributions on the simplex Σ. We denote by $\pi_0 = \{\pi_0^a\}_{a \in A} \in \prod_{a \in A} \Delta(\Sigma)$ the family of initial priors, one for each arm, and we formulate the following maximization problem — where the supremum is taken over $\mathbf{U} = \{\mathbf{U}_t^a\}_{a \in A, t \in [\![0,T-1]\!]} \in \{0,1\}^{A \times [\![0,T-1]\!]}$, subject to constraints (1) and (2),

$$V_0(\pi_0) = \sup \int_{\Delta(\Sigma)^A} \prod_{a \in A} \pi_0^a(\,\mathrm{d}p^a) \mathbb{E}_{\{p^a\}_{a \in A}} \left[\sum_{t=0}^{T-1} \sum_{a \in A} \mathbf{U}_t^a L_t^a(\mathbf{W}_{t+1}^a) \right] . \tag{3}$$

2.2 Dynamic programming and arm decomposition

Now, adapting the method in [4], we show how the stochastic optimal control problem (3) can be treated by decomposition.

Proposition 1. *For any vector $\mu = \{\mu_t\}_{t \in [\![0,T-1]\!]} \in \mathbb{R}^T$ of multipliers, we define the family $\{V_t^a[\mu]\}_{a \in A, t \in [\![0,T]\!]}$ of functions $V_t^a[\mu] : \mathbb{N} \times \mathbb{N} \to \mathbb{R}$ by the following backward induction: for all arm $a \in A$,*

$$\begin{aligned}
&V_T^a[\mu](n^{\mathtt{B}a}, n^{\mathtt{G}a}) = 0 \ , \ \ \forall (n^{\mathtt{B}a}, n^{\mathtt{G}a}) \in \mathbb{N} \times \mathbb{N} \ , \\
&V_t^a[\mu](n^{\mathtt{B}a}, n^{\mathtt{G}a}) = \max \Big\{ V_{t+1}^a[\mu](n^{\mathtt{B}a}, n^{\mathtt{G}a}), -\mu_t \\
&\qquad\qquad + \frac{n^{\mathtt{B}a}}{n^{\mathtt{B}a} + n^{\mathtt{G}a}} \big(L_t^a(\mathtt{B}) + V_{t+1}^a[\mu](n^{\mathtt{B}a} + 1, n^{\mathtt{G}a}) \big) \\
&\qquad\qquad + \frac{n^{\mathtt{G}a}}{n^{\mathtt{B}a} + n^{\mathtt{G}a}} \big(L_t^a(\mathtt{G}) + V_{t+1}^a[\mu](n^{\mathtt{B}a}, n^{\mathtt{G}a} + 1) \big) \Big\} \ , \\
&\qquad \forall (n^{\mathtt{B}a}, n^{\mathtt{G}a}) \in \mathbb{N} \times \mathbb{N} \ , \ \ \forall t \in [\![0, T-1]\!] \ .
\end{aligned} \tag{4}$$

Then, identifying (by an abuse of notation) $V_0\big((n_0^{\mathtt{B}a})_{a\in A},(n_0^{\mathtt{G}a})_{a\in A}\big)$ *with the value* $V_0(\pi_0)$ *of problem* (3) *when* $\pi_0=\{\beta(n_0^{\mathtt{B}a},n_0^{\mathtt{G}a})\}_{a\in A}$*, we have the upper bound*

$$V_0\big((n_0^{\mathtt{B}a})_{a\in A},(n_0^{\mathtt{G}a})_{a\in A}\big)\leq \inf_{\mu\in\mathbb{R}^T}\left(\sum_{a\in A}V_0^a[\mu](n_0^{\mathtt{B}a},n_0^{\mathtt{G}a})+\sum_{t=0}^{T-1}\mu_t\right). \tag{5}$$

Proof. The proof is in two steps. First, we transform the stochastic optimal control problem (3) under imperfect information into one under perfect state information. Second, we show how this latter can be decomposed, arm by arm, and we provide an upper bound.

It is well-known that a stochastic optimal control problem under imperfect information like (3) can be turned into a perfect state information (see [2, Chapter 10] for instance), but with the state being a probability distribution. For this purpose, we need to introduce some notation. For $\pi^a=\beta(n^{\mathtt{B}},n^{\mathtt{G}})$, we set $[\![\pi^a]\!]^{\mathtt{B}}=n^{\mathtt{B}}/(n^{\mathtt{B}}+n^{\mathtt{G}})$ and $[\![\pi^a]\!]^{\mathtt{G}}=n^{\mathtt{G}}/(n^{\mathtt{B}}+n^{\mathtt{G}})$. We also define the two shift mappings $\theta^{\mathtt{B}}\beta(n^{\mathtt{B}},n^{\mathtt{G}})=\beta(n^{\mathtt{B}}+1,n^{\mathtt{G}})$ and $\theta^{\mathtt{G}}\beta(n^{\mathtt{B}},n^{\mathtt{G}})=\beta(n^{\mathtt{B}},n^{\mathtt{G}}+1)$.

By [2, Propositions 10.5 and 10.6], it can be shown that the imperfect state information problem (3) can be reduced to a perfect state one, with information state $\pi_t=\{\pi_t^a\}_{a\in A}=\{\beta(n_t^{\mathtt{B}a},n_t^{\mathtt{G}a})\}_{a\in A}$, information state transition kernels

$$k_t(\,\mathrm{d}\pi_{t+1}\mid\pi_t,u_t)=\bigotimes_{a\in A}k_t^a(\,\mathrm{d}\pi_{t+1}^a\mid\pi_t^a,u_t^a)\,, \tag{6a}$$

$$\text{where}\quad k_t^a(\,\mathrm{d}\pi_{t+1}^a\mid\pi_t^a,u_t^a)=\begin{cases}\pi_t^a & \text{if } u_t^a=0\,,\\ [\![\pi_t^a]\!]^{\mathtt{B}}\delta_{\theta^{\mathtt{B}}\pi_t^a}+[\![\pi_t^a]\!]^{\mathtt{G}}\delta_{\theta^{\mathtt{G}}\pi_t^a} & \text{if } u_t^a=1\,,\end{cases} \tag{6b}$$

with the one-stage payoff

$$\tilde{L}_t(\pi_t,u_t)=\sum_{a\in A}u_t^a\Big([\![\pi_t^a]\!]^{\mathtt{B}}L_t^a(\mathtt{B})+[\![\pi_t^a]\!]^{\mathtt{G}}L_t^a(\mathtt{G})\Big)\,, \tag{7}$$

so that the original stochastic optimal control problem (3) under imperfect information is

$$V_0(\pi_0)=\sup\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{a\in A}\mathbf{U}_t^a\Big([\![\pi_t^a]\!]^{\mathtt{B}}L_t^a(\mathtt{B})+[\![\pi_t^a]\!]^{\mathtt{G}}L_t^a(\mathtt{G})\Big)\right] \tag{8a}$$

$$\pi_{t+1}^a\sim k_t^a(\,\mathrm{d}\pi_{t+1}^a\mid\pi_t^a,\mathbf{U}_t^a)\,,\ \ \forall a\in A\,, \tag{8b}$$

$$\mathbf{U}_t=\{\mathbf{U}_t^a\}_{a\in A}\in\{0,1\}^A\,, \tag{8c}$$

$$\sigma(\mathbf{U}_t)\subset\sigma(\{\pi_t^a\}_{a\in A})\,,\ \ \forall t\in[\![0,T-1]\!]\,, \tag{8d}$$

$$\sum_{a\in A}\mathbf{U}_t^a=1\,,\ \ \forall t\in[\![0,T-1]\!]\,. \tag{8e}$$

Now, adapting the method in [4], we show how the stochastic optimal control problem (8) can be treated by decomposition. For this purpose, we are going to dualize[4] the equality

[4]We have chosen the dualization term $-\mu_t\big(\sum_{a\in A}\mathbf{U}_t^a-1\big)$ and not $\mu_t\big(\sum_{a\in A}\mathbf{U}_t^a-1\big)$ because it is likely that, at the optimum, $\mu_t\geq 0$. Indeed, had we chosen the constraint $\sum_{a\in A}\mathbf{U}_t^a-1\leq 0$ (corresponding to selecting *at most* one arm, hence either no arm or a single arm), we would have considered multipliers $-\mu_t\leq 0$, hence $\mu_t\geq 0$.

constraints (8e). For any vector $\mu = \{\mu_t\}_{t\in[\![0,T-1]\!]} \in \mathbb{R}^T$ of multipliers, we readily get that (see (8a)) $\sum_{a\in A} \mathbf{U}_t^a \Big([\![\pi_t^a]\!]^{\mathtt{B}} L_t^a(\mathtt{B}) + [\![\pi_t^a]\!]^{\mathtt{G}} L_t^a(\mathtt{G}) \Big) - \mu_t \big(\sum_{a\in A} \mathbf{U}_t^a - 1 \big) =$
$\sum_{a\in A} \mathbf{U}_t^a \Big([\![\pi_t^a]\!]^{\mathtt{B}} L_t^a(\mathtt{B}) + [\![\pi_t^a]\!]^{\mathtt{G}} L_t^a(\mathtt{G}) - \mu_t \Big) + \mu_t$. Hence, from (8), we obtain

$$V_0(\pi_0) \leq \sup \sum_{a\in A} \sum_{t=0}^{T-1} \mathbb{E}\Big[\mathbf{U}_t^a \Big([\![\pi_t^a]\!]^{\mathtt{B}} L_t^a(\mathtt{B}) + [\![\pi_t^a]\!]^{\mathtt{G}} L_t^a(\mathtt{G}) - \mu_t \Big) \Big] + \sum_{t=0}^{T-1} \mu_t \tag{9a}$$

$$\pi_{t+1}^a \sim k_t^a(\, \mathrm{d}\pi_{t+1}^a \mid \pi_t^a, \mathbf{U}_t^a) \,, \ \ \forall a \in A \,, \tag{9b}$$

$$\mathbf{U}_t = \{\mathbf{U}_t^a\}_{a\in A} \in \{0,1\}^A \,, \tag{9c}$$

$$\sigma(\mathbf{U}_t) \subset \sigma(\{\pi_t^a\}_{a\in A}) \,, \ \ \forall t \in [\![0, T-1]\!] \,, \tag{9d}$$

and, because of the separability with respect to arms $a \in A$,

$$= \sum_{a\in A} \sup \mathbb{E}\Big[\sum_{t=0}^{T-1} \mathbf{U}_t^a \big([\![\pi_t^a]\!]^{\mathtt{B}} L_t^a(\mathtt{B}) + [\![\pi_t^a]\!]^{\mathtt{G}} L_t^a(\mathtt{G}) - \mu_t \big) \Big] + \sum_{t=0}^{T-1} \mu_t \tag{10a}$$

$$\pi_{t+1}^a \sim k_t^a(\, \mathrm{d}\pi_{t+1}^a \mid \pi_t^a, \mathbf{U}_t^a) \,, \tag{10b}$$

$$\mathbf{U}_t^a \in \{0,1\}^A \,, \tag{10c}$$

$$\sigma(\mathbf{U}_t^a) \subset \sigma(\pi_t^a) \,, \ \ \forall t \in [\![0, T-1]\!] \,, \tag{10d}$$

where the feedback constraint (9d) is reduced to (10d) because there is no loss of optimality as each inner maximization problem in (10) only depends on π_t^a, and not on the $\pi_t^{a'}$ for $a' \neq a$.

The corresponding dynamic programming equations, for the Bellman value functions $V_t^a[\mu]$ of each inner maximization problem in (10), are given by

$$\forall a \in A \,, \ \ \forall \pi^a \in \Delta(\Sigma) \,, \ \ V_T^a[\mu](\pi^a) = 0 \,, \ \ \forall t \in [\![0, T-1]\!] \,, \ \ V_t^a[\mu](\pi^a) =$$
$$\max_{u^a \in \{0,1\}} \Big(u^a \big([\![\pi^a]\!]^{\mathtt{B}} L_t^a(\mathtt{B}) + [\![\pi^a]\!]^{\mathtt{G}} L_t^a(\mathtt{G}) - \mu_t \big) + \int_\Sigma k_t^a(d\pi'^a \mid \pi^a, u^a) V_{t+1}^a[\mu](\pi'^a) \Big) \,.$$

As a consequence, by (6b) we obtain that $V_T^a[\mu](\pi^a) = 0$, $V_t^a[\mu](\pi^a) = \max\big\{ V_{t+1}^a[\mu](\pi^a),$ $[\![\pi^a]\!]^{\mathtt{B}} \big(L_t^a(\mathtt{B}) + V_{t+1}^a[\mu](\theta^{\mathtt{B}}\pi^a) \big) + [\![\pi^a]\!]^{\mathtt{G}} \big(L_t^a(\mathtt{G}) + V_{t+1}^a[\mu](\theta^{\mathtt{G}}\pi^a) \big) - \mu_t \big\}$. Thus, by (9) and (10), the optimal value $V_0(\pi_0)$ in (3) is such that $V_0(\pi_0) \leq \Big(\sum_{a\in A} V_0^a[\mu](\pi_0^a) + \sum_{t=0}^{T-1} \mu_t \Big)$, $\forall \mu \in \mathbb{R}^T$. Then, we readily get (5). □

3 The DeCo algorithm

To go beyond the limitations of the curse of dimensionality, we rely on the use of the decentralized control policy obtained by arm decomposition as described in §2.2. We call this algorithm DeCo (decomposition-coordination algorithm). By contrast to the (brute

force) dynamic programming solution (BF), in the decomposed formulation we have to solve Bellman equations for each arm, and thus we use Dynamic programming with a state of dimension 2, no matter the number of arms. The DeCo algorithm is made of an offline computation and of an online computation phases as follows. The offline phase is summarized in Figure 1: it consists in the minimization of a dual function φ, where each evaluation of φ relies on solving $|A|$ independent Bellman equations. The minimization step can be performed by gradient descent.

3.1 Offline phase of the DeCo algorithm

The offline phase of the DeCo algorithm is the minimization of the dual function $\varphi(\mu) = \left(\sum_{a\in A} V_0^a[\mu](\pi_0^a) + \sum_{t=0}^{T-1} \mu_t\right)$, for all $\mu \in \mathbb{R}^T$, for a family $\{\pi_0^a\}_{a\in A} = \{\beta(n_0^{\mathtt{B}a}, n_0^{\mathtt{G}a})\}_{a\in A}$ of beta priors. The algorithm is summarized in Figure 1 and its four steps are as follows.

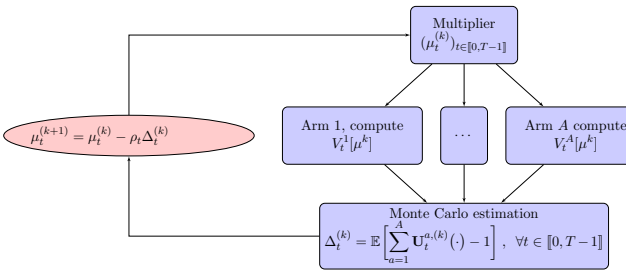

Figure 1: The decomposition coordination algorithm (DeCo)

- (S_1) Choose an initial vector $\mu^{(0)} \in \mathbb{R}^T$ of multipliers.
- (S_2) At step k, given a vector $\mu^{(k)} \in \mathbb{R}^T$ of multipliers, compute the collection $\left\{V_t^a[\mu^{(k)}]\right\}_{t\in[\![0,T]\!],a\in A}$ of Bellman value functions given by (4) and the collection of associated optimal controls. The computation is performed in parallel, arm per arm. Note that $V_t^a[\mu^{(k)}]$, the Bellman value function at time $t \in [\![0, T]\!]$, is to be evaluated only on the finite grid $\{(n_0^{\mathtt{B}a} + n^{\mathtt{B}a}, n_0^{\mathtt{G}a} + n^{\mathtt{G}a}) \,|\, n^{\mathtt{B}a} + n^{\mathtt{G}a} \leq t\}$. Note also that, when all the arms share the same prior and the same instantaneous reward, a unique sequence of Bellman value functions is to be computed, that is, all the arms share the same sequence of Bellman value functions.
- (S_3) Once gotten the subfamily $\left\{V_0^a[\mu^{(k)}]\right\}_{a\in A}$ of initial Bellman value functions at step k, update the vector of multipliers by a gradient step to obtain $\mu^{(k+1)}$. The gradient of the dual function φ with respect to the multipliers is obtained by computing the expectation of the dualized constraint as formulated in Problem (9) (see [5] for more details). Numerically, the expectation is obtained by Monte Carlo simulations. The gradient step can be replaced by a more sophisticated algorithm such as the conjugate gradient or the quasi-Newton method. In our numerical experiments, we use a solver (limited memory Bfgs) of the Modulopt library from Inria [11].

(S_4) Stop the iterations (stopping criterion) or go back to Item S_2 with multiplier $\mu^{(k+1)}$.

3.2 Online phase of the DeCo algorithm

The stochastic optimal control problem (8) is, theoretically, solvable by dynamic programming. Denoting by $\{V_t\}_{t\in[\![0,T]\!]}$ the corresponding Bellman value functions, an optimal policy would be given by the feedback (where $\pi_t = \{\pi_t^a\}_{a\in A} = \{\beta(n_t^{\mathtt{B}a}, n_t^{\mathtt{G}a})\}_{a\in A}$)

$$U_t(\pi_t) \in \underset{\substack{u_t=\{u_t^a\}_{a\in A}\in\{0,1\}^A \\ \sum_{a\in A} u_t^a = 1}}{\arg\max} \Big(\tilde{L}_t(\pi_t, u_t) + \int_{\Delta(\Sigma)} V_{t+1}(\pi_{t+1}) k_t(\,\mathrm{d}\pi_{t+1} \mid \pi_t, u_t) \Big) \,. \tag{11}$$

The DeCo algorithm consists in replacing the Bellman value function V_{t+1} by $\sum_{a\in A} V_{t+1}^a[\mu]$, using the collection $\big\{V_{t+1}^a[\mu]\big\}_{a\in A}$, of Bellman value functions given by (4), and a suitable vector $\mu \in \mathbb{R}^T$.

Using the expressions (6a)–(6b) for the kernels $k_t(\,\mathrm{d}\pi_{t+1} \mid \pi_t, u_t)$ and (7) for the new one-stage payoff $\tilde{L}_t(\pi_t, u_t)$, an easy computation (using $\sum_{\substack{a'\in A \\ a'\neq a}} V_{t+1}^{a'}[\mu] = \sum_{a'\in A} V_{t+1}^{a'}[\mu] - V_{t+1}^a[\mu]$) gives the following policy. When the state of the multi-armed system is given by $(n_t^{\mathtt{B}a}, n_t^{\mathtt{G}a})_{a\in A} \in \prod_{a\in A} \mathbb{N}\times\mathbb{N}$ at time t, the DeCo algorithm selects an arm $\mathcal{A}^*[\mu]\big(\{(n_t^{\mathtt{B}a}, n_t^{\mathtt{G}a})\}_{a\in A}\big)$ in[5]

$$\begin{aligned} \underset{a\in A}{\arg\max} \Bigg(& - V_{t+1}^a[\mu](n_t^{\mathtt{B}a}, n_t^{\mathtt{G}a}) + \frac{n_t^{\mathtt{B}a}}{n_t^{\mathtt{B}a} + n_t^{\mathtt{G}a}} \big(L_t^a(\mathtt{B}) + V_{t+1}^a[\mu](n^{\mathtt{B}a}{+}1, n^{\mathtt{G}a}) \big) \\ & + \frac{n_t^{\mathtt{G}a}}{n_t^{\mathtt{B}a} + n_t^{\mathtt{G}a}} \big(L_t^a(\mathtt{G}) + V_{t+1}^a[\mu](n^{\mathtt{B}a}, n^{\mathtt{G}a}{+}1) \big) \Bigg) \,. \end{aligned} \tag{12}$$

The structure of policy (12) is that of a *nonstationary*[6] *index policy*. Indeed, the right hand side in (12) is a quantity that depends only on t and on the state $(n_t^{\mathtt{B}a}, n_t^{\mathtt{G}a})$ of arm a at time t. The DeCo policy used in numerical experiments is the policy $\mathcal{A}^*[\mu^*]$ in (12), where μ^* is given by the offline phase of the DeCo algorithm.

4 Numerical experiments

In this section, we present numerical results. The policies $\mathbf{U} = \{\mathbf{U}_t^a\}_{a\in A, t\in[\![0,T-1]\!]}$ are compared using the expected Bayesian regret given by

$$\mathcal{R}(\mathbf{U}) = \int_{\Delta(\Sigma)^A} \prod_{a\in A} \pi_0^a(\,\mathrm{d}p^a) \mathbb{E}_{\{p^a\}_{a\in A}} \Bigg[\sum_{t=0}^{T-1} \sum_{a\in A} \big(\mathbf{U}_t^{\mathrm{BA},a} - \mathbf{U}_t^a \big) \mathbf{W}_{t+1}^a \Bigg] \,, \tag{13}$$

where we have set the instantaneous costs L_t^a equal to 1 on $\mathtt{G}$ and 0 on $\mathtt{B}$ for and where the BA (best arm) policy is, for all $a \in A$, given by $\mathbf{U}_t^{\mathrm{BA},a} = 1 \iff a \in \arg\max_{a'\in A} p_{\mathtt{G}}^{a'}$, and

[5] In case of non uniqueness, take any arm in the arg max.

[6] If we had considered an infinite horizon, we would have obtained a (stationary) index policy.

where the prior is supposed to be the uniform law for all arms. As an example the DeCo policy is defined, for all $a \in A$, by $\mathbf{U}_t^{\text{DeCo},a} = 1 \iff a \in \mathcal{A}^*[\mu]\big(\{(n_t^{\text{B}a}, n_t^{\text{G}a})\}_{a\in A}\big)$, where $\mathcal{A}^*[\mu]$ is defined in (12). See Footnote 5 in case of non uniqueness.

4.1 Algorithms tested

The DeCo algorithm The expected Bayesian regret (13) is evaluated using the DeCo policy detailed above. Numerically, the expected Bayesian regret is obtained by Monte Carlo simulations, where the expectation with respect to the prior is obtained with a sample of size 1000 and expectation with respect to the arms parameters is obtained with a sample of size 1000 in Figure 2 and of size 100 in Figure 3.

Moreover, the Bellman upper bound given by the right hand side of Inequality (5), associated with the multiplier μ^*, yields the inequality

$$\mathcal{R}(\mathbf{U}) \geq \mathcal{R}^{\text{LB}} = \frac{|A|}{|A|+1}(T-1) - \left(\sum_{a\in A} V_0^a[\mu^*](\pi_0^a) + \sum_{t=0}^{T-1} \mu_t^*\right). \tag{14}$$

The lower bound $\mathcal{R}^{\text{LB}}$, for the expected Bayesian regret (13), will then be compared to the expected Bayesian regret obtained by the following policies (DeCo, Ts, Kl-Ucb and, possibly, to the exact value given by Bf).

The Ts and the Kl-Ucb algorithms The Ts and Kl-Ucb policies are index policies. The associated expected Bayesian regret values (13) are obtained, as for the DeCo policy, by Monte Carlo simulations. It should be noted that the Monte Carlo samples are the same for all the evaluated policies.

Brute force Dynamic programming (Bf) Here, we do not describe a policy $\mathbf{U} = \{\mathbf{U}_t^a\}_{a\in A, t\in[\![0,T-1]\!]}$, but an algorithm Bf to compute $V_0(\pi_0)$ in (3). Solving the maximization problem (3), that is, computing $V_0(\pi_0)$ for a given prior (like, for instance, the uniform law given by the beta distribution $\beta(1,1)$ for all arms) can be done using Dynamic programming on the equivalent formulation (8). This is however only possible for relatively small instances of problem (3), that is, for a limited number $|A|$ of arms and a limited time horizon T. We recall here that solving the problem for $|A|$ arms requires solving a Bellman equation with a state of dimension $2|A|$ (a state described by two integers per arm), which implies an exponential increase in computational cost with respect to $|A|$. This is an instance of what Richard Bellman referred to as the *curse of dimensionality*.

4.2 Results

For the sake of reproducibility, we have performed two separate implementations with two different languages, one in Python 3, the other in Nsp [7].

Comparison with the optimal solution on simple instances In Table 1, we compare the performance of the DeCo policy against the brute force approach Bf. As already explained, such comparison is limited by the computational cost of the brute force method. The results, expressed in term of total expected reward, are derived from the Bellman value functions — the solution of the recursion— for the Bf simulation and are computed by sample average (Monte Carlo simulations) for the DeCo policy (see §3 for details). We observe that the performance of the decentralized policy DeCo is close to the optimal solution while keeping the computational cost reasonable (at most 1.3 second) when the number of arms increase. As the computation of the expected Bayesian regret (13) by Monte Carlo

Table 1: Comparison of Bf and DeCo in term of total expected reward (higher is better). As, for DeCo, this quantity is estimated with Monte-Carlo simulation (see §4.1), it might happen that the empirical average makes better than Bf, but this is due to the simulation noise. For those examples for which a resolution with Bf is feasible, we observe that DeCo is very close to optimality.

$\|A\|$	T	Bf	DeCo
2	5	2.888	2.892
2	20	12.431	12.436
2	50	31.996	31.872

$\|A\|$	T	Bf	DeCo
3	10	6.409	6.411
3	20	13.465	13.458
5	10	6.659	6.645

simulations is computationaly expensive, scaling the computation, as the horizon increases, is left for further work.

Comparison of the regrets We also compare the DeCo policy against the Thompson Sampling policy Ts and Kullback-Leibler upper-confidence bound Kl-Ucb [10]. Note that all policies use a uniform prior as initial state.

The results are provided in Figure 2. On all cases, DeCo beats both Ts and Kl-Ucb with a comfortable margin. For the two arms case in Figure 2(a), DeCo is very close to the optimal solution, computed by dynamic programming (we used the Julia BinaryBandit library [14, 13]). Also, we observe that the lower bound (14), obtained on the regret by adapting Proposition 1, becomes very close to the regret obtained with DeCo for the experiment with 20 arms in Figure 2(d). This means that, on this instance, we are close to the optimal solution one would get with dynamic programming (which is not feasible for 20 arms).

Figure 3 shows the regret lower bound, and the DeCo, Ts and Kl-Ucb regrets as a function of the number of arms for $T = 100$ and $T = 500$ (beware: the x axis is the number of arms!). The lower bound is of no use (lower than 0) for small numbers 2 and 5 of arms. Nevertheless, when the number of arms increases, the regret of DeCo and the lower bound become quite close, which indicates that, for those examples, DeCo is close to being optimal.

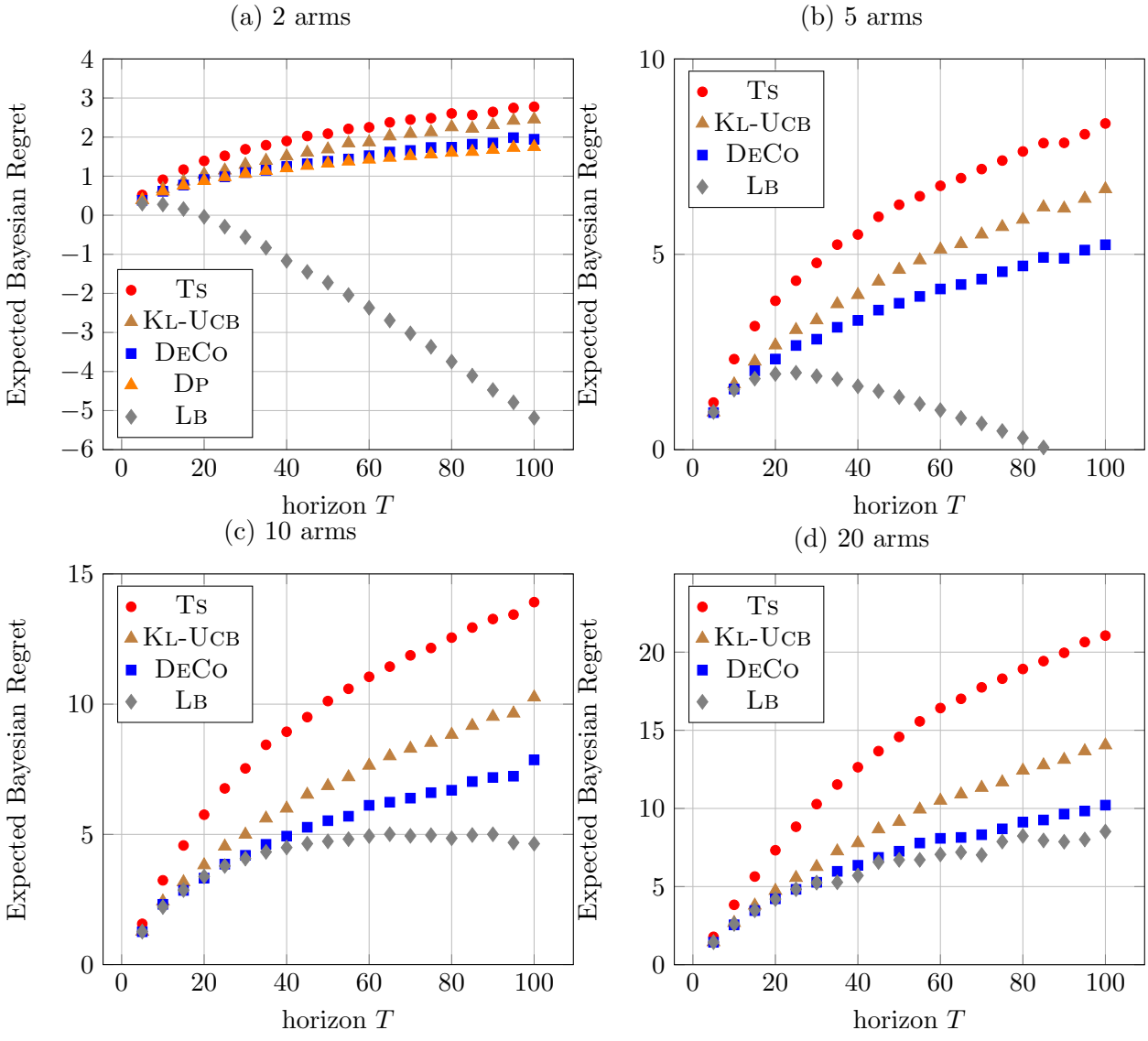

Figure 2: Expected Bayesian regret (13) for DECO, TS and KL-UCB policies (the lower the better) for 2, 5, 15 and 20 arms with uniform prior. The (DECO) lower bound LB in (14) is also plotted.

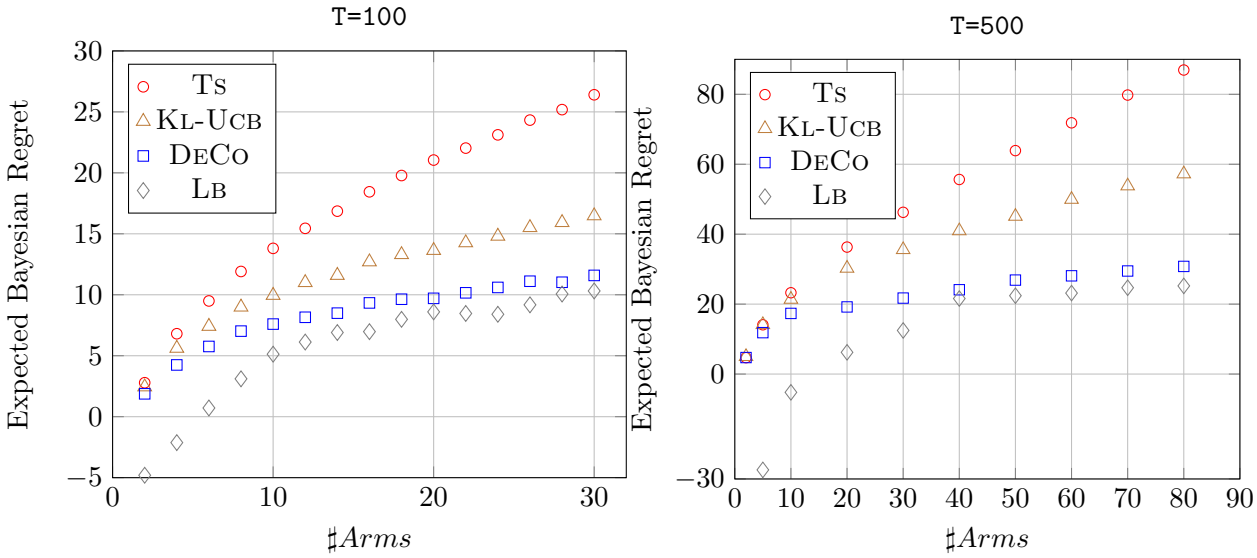

Figure 3: Expected Bayesian regret (13) for DeCo, Ts and Kl-Ucb with uniform prior, as functions of the number of arms. The (DeCo) lower bound Lb in (14) is also plotted and demonstrates that DeCo is close to the optimal solution when the number of arms is large enough. Here, the expectation with respect to the prior is obtained with samples of size 1000 and expectation with respect to the arms parameters is obtained with samples of size 100 (compared to 1000 in Figure 2).

5 Conclusion

The numerical results demonstrate the value of the decomposition-coordination approach: DeCo performances are close to the optimal Bayesian solution for several configurations of arms and horizons, while keeping the computing time reasonable. Further works include extensions to the discounted infinite horizon case, to the frequentist setting, as well as the theoretical analysis of the DeCo policy. Also, while we have not reproduced their results, the DeCo policy seems to have performance close to the state of the art [19, 9].

References

[1] R. Bellman. A problem in the sequential design of experiments. *Sankhyā: The Indian Journal of Statistics (1933-1960)*, 16(3/4):221–229, 1956.

[2] D. P. Bertsekas and S. E. Shreve. *Stochastic Optimal Control: The Discrete-Time Case*. Athena Scientific, Belmont, Massachusetts, 1996.

[3] S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*, 2012.

[4] P. Carpentier, J.-P. Chancelier, M. De Lara, and F. Pacaud. Mixed spatial and temporal decompositions for large-scale multistage stochastic optimization problems. *Journal of Optimization Theory and Applications*, 186(3):985–1005, 2020.

[5] P. Carpentier, J.-P. Chancelier, V. Leclère, and F. Pacaud. Stochastic decomposition applied to large-scale hydro valleys management. *European Journal of Operational Research*, 270(3):1086–1098, 2018.

[6] J. Chakravorty and A. Mahajan. Multi-armed bandits, Gittins index, and its calculation. *Methods and applications of statistics in clinical trials: Planning, analysis, and inferential methods*, 2(416-435):455, 2014.

[7] J.-P. Chancelier. Website: `http://cermics.enpc.fr/~jpc/nsp-tiddly`. Nsp, a numerical computing environment, 2021.

[8] O. Chapelle and L. Li. An empirical evaluation of thompson sampling. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

[9] V. F. Farias and E. Gutin. Optimistic Gittins indices. In *Proceedings of the 30th International Conference on Neural Information Processing Systems,(3161-3169)*, 2016.

[10] A. Garivier and O. Cappé. The KL-UCB algorithm for bounded stochastic bandits and beyond. In *Proceedings of the 24th annual conference on learning theory*, pages 359–376. JMLR Workshop and Conference Proceedings, 2011.

[11] J. C. Gilbert and X. Jonsson. LIBOPT – An environment for testing solvers on heterogeneous collections of problems, 2007.

[12] J. Gittins, K. Glazebrook, and R. Weber. *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.

[13] P. Jacko. Binarybandit: An efficient julia package for optimization and evaluation of the finite-horizon bandit problem with binary responses, 2019.

[14] P. Jacko. The finite-horizon two-armed bandit problem with binary responses: A multidisciplinary survey of the history, state of the art, and myths, 2019.

[15] T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[16] T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[17] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

[18] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

[19] D. Russo and B. Van Roy. Learning to optimize via information-directed sampling. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[20] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[21] J. Weed, V. Perchet, and P. Rigollet. Online learning in repeated auctions. In *Conference on Learning Theory*, pages 1562–1583. PMLR, 2016.