When Gaussian Processes Meet Combinatorial Bandits: GCB

Guglielmo Maria Accabi GUGLIELMO.ACCABI@MAIL.POLIMI.IT
Francesco Trovò FRANCESCO1.TROVO@POLIMI.IT
Alessandro Nuara ALESSANDRO.NUARA@POLIMI.IT
Nicola Gatti NICOLA.GATTI@POLIMI.IT
Marcello Restelli MARCELLO.RESTELLI@POLIMI.IT
Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano, Milano, 20133, Italy

Abstract

Combinatorial bandits (CMAB) are a generalization of the well-known Multi-Armed Bandit framework, in which the learner chooses, at each round, a subset of the available arms that satisfies some known constraints. The learner observes the payoffs of each chosen arm and aims at maximizing the cumulative reward. We study, for the first time, CMAB settings with some form of correlation over the arms expected rewards. The arm correlation is crucial to allow algorithms to be effective when the space of the arms is large. In the present paper, we propose a bandit algorithm, namely Gaussian Combinatorial Bandit (GCB), designed for settings in which the arms are partitioned in subsets, and the payoff functions of the arms of each subset are jointly distributed as a Gaussian Process (GP). We provide two different variations of our algorithm (frequentist and Bayesian) that, under mild assumptions, their worst-case regret is $\tilde{\mathcal{O}}(C\sqrt{N})$, where C is the number of subsets of arms whose payoffs are correlated and N is the number of rounds.[1]

1. Introduction

The Multi-Armed Bandit (MAB) settings (Bubeck et al., 2012) have been proven to be a powerful tool to tackle real-world problems in which decisions are taken sequentially. In these settings, a learner chooses, at each round, an option among a set of available ones, usually called *arms*, and she receives a stochastic *reward* associated with the chosen arm. The learner aims at minimizing the expected loss, called *regret*, incurred due to the lack of *a priori* information on the arm maximizing the expected reward. Thanks also to their successful adoption in many real-world applications, MAB algorithms have been achieving great popularity in the last years. In our work, we focus on a class of problems called Combinatorial MAB (CMAB). The peculiarity of CMAB is that the learner chooses at each round a subset of arms, called *super-arms*, subject to a set of constraints (e.g., knapsack constraints), observes the payoffs of every single arm belonging to the chosen super-arm, and gets the corresponding reward. Usually, the constraints have combinatorial nature, thus making the optimization problem NP-hard. The CMAB algorithms developed so far, e.g., by (Chen et al., 2013), (Gai et al., 2010), (Shleyfman et al., 2014), (Ontañón, 2017), aim at exploiting the correlation among the super-arms due to the potential non-null intersection of the subsets of arms of different super-arms. These algorithms are then paired with exact or approximation oracles solving the optimization problem.

[1] With the notation $\tilde{\mathcal{O}}$ we disregard logarithmic factors.

In several challenging applications, the space of the arms of a CMAB setting may be large, making the learning process inefficient unless some form of correlation among the expected payoffs can be exploited. This happens, e.g., in the optimization of an Internet advertising campaign (Nuara et al., 2018), in which the problem consists in finding the bid/daily budget pair for every subcampaign maximizing the number of total daily conversions (e.g., clicks or purchases) s.t. the total daily spent is not larger than a given budget. For every subcampaign, there is a set of arms representing a discretization of the 2-dimensional bid/daily budget joint space. This space presents some form of regularity, as the number of conversions changes smoothly when the bid or/and the daily budget vary. Another application fitting the CMAB framework is the allocation of cells in a 5G network (Maghsudi and Hossain, 2016). The problem consists in powering a subset of cells to provide a fast connection to the largest portion of users, while keeping the energy costs under a threshold and the correlation is induced by the spatial nature of the problem.

In the present paper, we provide the first CMAB algorithm with theoretical guarantees exploiting the correlation existing among the expected payoffs of the arms. In particular, we study a specific CMAB setting, that we call Gaussian Process-CMAB (GP-CMAB), in which the space of the arms is partitioned in subsets of arms such that the payoff functions of the arms of every subset are jointly distributed as a Gaussian Process (GP). We propose two variations of the GCB algorithm, namely GCB-UCB and GCB-TS, based on the use of upper confidence bounds and a posterior sampling procedure, respectively. For each variation of our algorithm, we provide a high-probability regret of $\tilde{\mathcal{O}}(C\sqrt{N})$, where C is the number of the subsets of correlated arms and N is the number of rounds.

2. The GP-CMAB Setting

A *GP-CMAB problem* consists of a finite set $\mathcal{D}$ of M arms, which is partitioned into C disjoint subsets of arms $\mathcal{D}_1, \ldots, \mathcal{D}_C$. Each subset $\mathcal{D}_i := \{\boldsymbol{a}_{i1}, \ldots, \boldsymbol{a}_{iM_i}\}$ is composed of $M_i \in \mathbb{N}$ arms $\boldsymbol{a}_{ij} \in \mathbb{R}^d$, where $d \in \mathbb{N}^+$ is the *dimension* of each arm. Each subset $\mathcal{D}_i$ is characterized by an expected payoff function $\mu_i : \mathcal{D}_i \to \mathbb{R}$, which is the realization of a GP (Rasmussen and Williams, 2006). At every round t over a finite time horizon N, the learner pulls a *super-arm* $S_t \in \mathcal{S}$, where the set $\mathcal{S} \subseteq 2^{\mathcal{D}}$ is a subset of the power set of $\mathcal{D}$ that satisfies some constraints on the super-arm composition. Once the learner has chosen the super-arm S_t, she observes a noisy realization from the payoff function $y_t(\boldsymbol{a}) := \mu_i(\boldsymbol{a}) + \varepsilon_t(\boldsymbol{a})$ for each arm $\boldsymbol{a} \in S_t \cap \mathcal{D}_i$ and for each $i \in \{1, \ldots C\}$, where $\varepsilon_t(\boldsymbol{a}) \sim \mathcal{N}(0, \sigma^2)$ for each t and $\boldsymbol{a}$, and $\sigma^2 \in \mathbb{R}^+$. Finally, she earns a *reward* $R_t(S_t) = f(S_t, \{y_t(\boldsymbol{a})\}_{\boldsymbol{a} \in S_t})$ which depends on the chosen super-arm S_t and on the payoffs $y_t(\boldsymbol{a})$. In the simplest case, f is the sum of the payoffs of the arms in S_t, i.e., $R_t(S_t) = \sum_{\boldsymbol{a} \in S_t} y_t(\boldsymbol{a})$ (the modeling also allows for more complex definitions of the reward). Let us define $r_{\boldsymbol{\mu}}(S) := \mathbb{E}[R_t(S)]$ as the expected reward of a super-arm S and $\boldsymbol{\mu} := (\mu_{11}, \ldots, \mu_{CM_C})$, $\boldsymbol{\mu} \in \mathbb{R}^M$, with $\mu_{ij} := \mu_i(\boldsymbol{a}_{ij})$, as the vector of the expected values of the payoffs. We assume that the following properties hold.

Assumption 1 (Monotonicity) *The expected reward $r_{\boldsymbol{\mu}}(S)$ is monotonically non decreasing in $\boldsymbol{\mu}$, i.e., given $\boldsymbol{\mu}, \boldsymbol{\eta} \in \mathbb{R}^M$ s.t. $\mu_{ij} \leq \eta_{ij}, \forall (i,j)$ we have $r_{\boldsymbol{\mu}}(S) \leq r_{\boldsymbol{\eta}}(S)\ \forall\, S \in \mathcal{S}$.*

Assumption 2 (Lipschitz continuity) *The expected reward $r_{\boldsymbol{\mu}}(S)$ is Lipschitz continuous in the infinite norm w.r.t. the expected payoff vector $\boldsymbol{\mu}$, with Lipschitz constant $\Lambda > 0$.*

Formally, for each $\boldsymbol{\mu}, \boldsymbol{\eta} \in \mathbb{R}^M$ we have $|r_{\boldsymbol{\mu}}(S) - r_{\boldsymbol{\eta}}(S)| \leq \Lambda ||\boldsymbol{\mu} - \boldsymbol{\eta}||_\infty$, where the infinite norm of an expected payoff vector is $||\boldsymbol{\mu}||_\infty := \max_{i \in \{1,\ldots,C\}} \max_{j \in \{1,\ldots,M_i\}} |\boldsymbol{\mu}_{ij}|$.

These two assumptions assure that the expected reward $r_{\boldsymbol{\mu}}(S)$ does not decrease when we increase at least one of elements in the expected payoffs vector and that $r_{\boldsymbol{\mu}}(S)$ is Lipschitz continuous w.r.t. the expected payoff vector $\boldsymbol{\mu}$. The goal of the learner is to find the optimal expected reward $r^*_{\boldsymbol{\mu}} = \max_{S \in \mathcal{S}} r_{\boldsymbol{\mu}}(S)$. We assume that the learner has access to an *approximation oracle* able to solve approximately this optimization problem.

Definition 1 (**(α, β)-*Approximation Oracle***) *Given $\alpha, \beta \in [0, 1]$, an (α, β)-Approximation Oracle $\widehat{S} = \mathsf{Oracle}(\boldsymbol{\mu})$, with input the payoff vector $\boldsymbol{\mu}$ and output the super-arm $\widehat{S} \in \mathcal{S}$:*

$$\mathbb{P}\left[r_{\boldsymbol{\mu}}(\widehat{S}) \geq \alpha \, r^*_{\boldsymbol{\mu}}\right] \geq \beta.$$

A policy $\mathfrak{U}$ is an algorithm that selects a super-arm S_t at round t with the aim to minimize the loss due to the learning process. Since we only assume to have an approximation oracle, we need to define the expected pseudo-regret taking into account the suboptimality of the solution returned by the oracle as follows:

Definition 2 (**(α, β)-*Approximation Pseudo-Regret***) *Given the expected payoff vector $\boldsymbol{\mu}$ and an (α, β)-Approximation Oracle, the (α, β)-Approximation Pseudo-Regret $\mathcal{R}_N(\mathfrak{U})$ after N rounds of a given policy $\mathfrak{U}$ that selects the super-arm S_t at round t is defined as:*

$$\mathcal{R}_N(\mathfrak{U}) = N \, \alpha \, \beta \, r^*_{\boldsymbol{\mu}} - \mathbb{E}\left[\sum_{t=1}^{N} r_{\boldsymbol{\mu}}(S_t)\right].$$

We assume $\boldsymbol{\mu}$ to be unknown to the learner, so the policy $\mathfrak{U}$ needs to build estimates from the GPs to find a good approximation of $\boldsymbol{\mu}$ and of $r_{\boldsymbol{\mu}}(S)$ for any super-arm $S \in \mathcal{S}$, while keeping the pseudo-regret $\mathcal{R}_N(\mathfrak{U})$ due to this exploration as small as possible.

3. Gaussian Processes and Information Gain

A Gaussian Process $GP(\mu_i(\boldsymbol{a}), k_i(\boldsymbol{a}, \boldsymbol{a}'))$ is a collection of random variables, whose law is a multivariate Gaussian. It is specified by its mean function $\mu_i(\boldsymbol{a})$ and covariance (or kernel) $k_i(\boldsymbol{a}, \boldsymbol{a}') \leq 1$, where $\boldsymbol{a}, \boldsymbol{a}'$ are elements of $\mathcal{D}_i$. We focus on the following kernels, being the most common: **linear kernel** $k_i(\boldsymbol{a}, \boldsymbol{a}') = \boldsymbol{a}^\mathsf{T} \boldsymbol{a}'$; **squared exponential kernel** $k_i(\boldsymbol{a}, \boldsymbol{a}') = \exp\left\{-\frac{||\boldsymbol{a} - \boldsymbol{a}'||_2^2}{2 l_i^2}\right\}$, where l_i is a lengthscale; **Matérn kernel** $k_i(\boldsymbol{a}, \boldsymbol{a}') = (2^{1-\nu_i}/\Gamma(\nu_i)) r^{\nu_i} B_{\nu_i}(r)$, with $r = \frac{\sqrt{2\nu_i}}{w} ||\boldsymbol{a} - \boldsymbol{a}'||_2$, where $B_{\nu_i}(r)$ is the modified Bessel function of the 2^{nd} kind, $\Gamma(\nu_i)$ is the gamma function and $\nu_i \in \mathbb{R}^+$ is a smoothness parameter.

In what follows, we use GPs to approximate the unknown payoff functions μ_i. More specifically, we assume a prior distribution of $GP(0, k_i(\boldsymbol{a}, \boldsymbol{a}'))$ for an unknown function, which allow us to compute the posterior distribution by an analytic solution. Formally, we have that, given a sequence of t observations of the payoff function $\boldsymbol{y}(A_{i,t}) := [\mu_i(\boldsymbol{a}_{i,1}) + \varepsilon_1, \ldots, \mu_i(\boldsymbol{a}_{i,t}) + \varepsilon_t]^\mathsf{T}$ corresponding to the sequence of the chosen arms $A_{i,t} := [\boldsymbol{a}_{i,1}, \ldots, \boldsymbol{a}_{i,t}]$, where $\boldsymbol{a}_{i,h} \in \mathcal{D}_i$ is an arm from the subset $\mathcal{D}_i$ pulled at round h and $\varepsilon_h \sim \mathcal{N}(0, \sigma^2)$ are uncorrelated Gaussian noises, the posterior mean and variance for $\boldsymbol{a} \in \mathcal{D}_i$ are:

Algorithm 1: GCB

```
   Data: Set of arms $\mathcal{D}$, noise variance $\sigma^2$, GP Prior distributions $\hat{\mu}_{i,0}$ and $\hat{\sigma}^2_{i,0}$ for all $i \in \{1, \dots, C\}$
1  for $t \in \{1, \dots, N\}$ do
2      for $i \in \{1, \dots, C\}$ do
3          for $\boldsymbol{a} \in \mathcal{D}_i$ do
4              Compute $\hat{\mu}_{i,T_i(t-1)}(\boldsymbol{a})$ as in Equation (1)
5              Compute $\hat{\sigma}^2_{i,T_i(t-1)}(\boldsymbol{a})$ as in Equation (2)
6              Compute $u_{i,T_i(t-1)}(\boldsymbol{a})$ as in Equation (3) or $\theta_{i,T_i(t-1)}(\boldsymbol{a})$ as in Equation (4)
7      Generate the vector $\bar{\boldsymbol{\mu}}_t$
8      Select super-arm $S_t = \mathsf{Oracle}(\bar{\boldsymbol{\mu}}_t)$
9      Observe the payoffs $y_t(\mathbf{a})$ for each $\mathbf{a} \in \mathbf{S_t}$
10     for $i \in \{1, \dots, C\}$ do
11         Update the sequences $\boldsymbol{y}(A_{i,T_i(t-1)})$ and $A_{i,T_i(t-1)}$, if necessary
```

$$\hat{\mu}_{i,t}(\boldsymbol{a}) = \boldsymbol{k}_{i,t}(\boldsymbol{a})^\mathsf{T}(\boldsymbol{K}_{i,t} + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{y}(A_{i,t}), \tag{1}$$

$$\hat{\sigma}^2_{i,t}(\boldsymbol{a}) = k_i(\boldsymbol{a}, \boldsymbol{a}) - \boldsymbol{k}_{i,t}(\boldsymbol{a})^\mathsf{T}(\boldsymbol{K}_{i,t} + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{k}_{i,t}(\boldsymbol{a}), \tag{2}$$

where $\boldsymbol{k}_{i,t}(\boldsymbol{a}) := [k_i(\boldsymbol{a}_{i,1}, \boldsymbol{a}), \dots, k_i(\boldsymbol{a}_{i,t}, \boldsymbol{a})]^\mathsf{T}$ is the vector of the covariance between $\boldsymbol{a}$ and the arms in $A_{i,t}$, $\boldsymbol{K}_{i,t}$ is the Gram matrix and $\boldsymbol{I}$ is the identity matrix of order t.

The problem of selecting the sequence of arms $A_{i,t}$ whose payoff realizations $\boldsymbol{y}(A_{i,t})$ maximize the information on this function is already known in the literature as *Bayesian experimental design* (Chaloner and Verdinelli, 1995), where the *Information Gain* is used to measure how much information on μ_i is gained by sampling a sequence of arms $A_{i,t}$. Given the vector $\boldsymbol{y}(A_{i,t})$, Srinivas et al. (2010) show that the Information Gain in the case μ_i is a realization of a GP is defined as $I(\boldsymbol{y}(A_{i,t}) \mid \mu_i) := \frac{1}{2} \log |\boldsymbol{I} + \sigma^{-2} \boldsymbol{K}_{i,t}|$. Therefore, our problem is equivalent to finding a sequence of arms $A_{i,t}$ s.t. $\boldsymbol{a}_{i,h} \in \mathcal{D}_i$ for each $h \in \{1, \dots, t\}$ of length t providing the *maximum information gain* $\gamma_{i,t} := \max_{A_{i,t} | \boldsymbol{a}_{i,h} \in \mathcal{D}_i} I\left(\boldsymbol{y}(A_{i,t}) \mid \mu_i\right)$. However, the Greedy procedure proposed by Ko et al. (1995) to solve this problem does not provide guarantees on the pseudo-regret $\mathcal{R}_N(\mathfrak{U})$.

4. The GCB Algorithm

Our algorithm, called *Gaussian Combinatorial Bandit* (GCB), employs a set of GPs to estimate the payoff functions μ_i with $i \in \{1, \dots, C\}$. Then, the estimated payoffs for each arm in $\mathcal{D}$ are fed to the approximation oracle which chooses the super-arm S_t to play at round t. The pseudo-code of GCB is provided in Algorithm 1. The algorithm requires as input the set of arms $\mathcal{D}$, the time horizon N and a prior for each one of the GPs μ_i specified by the mean function $\hat{\mu}_{i,0}$ and the covariance function $\hat{\sigma}^2_{i,0}$. At round t, the algorithm computes estimates for the expected payoff for each arm $\boldsymbol{a} \in \mathcal{D}$ (Lines 4-5). To do so, the algorithm relies on the observed payoff vector $\boldsymbol{y}(A_{i,T_i(t-1)})$ corresponding to the arms in $A_{i,T_i(t-1)}$ selected during the previous rounds, where $T_i(t-1)$ is the number of arms observed up to round $t-1$ in the set $\mathcal{D}_i$. Using $\boldsymbol{y}(A_{i,T_i(t-1)})$, the posterior of the GP corresponding to the subset $\mathcal{D}_i$ for an arm $\boldsymbol{a} \in \mathcal{D}_i$ is a Gaussian distribution with mean $\hat{\mu}_{i,T_i(t-1)}(\boldsymbol{a})$ as in Equation (1) and variance $\hat{\sigma}^2_{i,T_i(t-1)}(\boldsymbol{a})$ as in Equation (2). Such a model provides a probability distribution for each expected payoff, which is not directly employable in the approximation oracle, that, instead, needs a single value per expected payoff vector. We cope with this issue using two different approaches (Line 6): the first follows the frequentist framework computing an upper bound of the mean reward $u_{i,T_i(t-1)}(\boldsymbol{a})$, while the second one follows the Bayesian framework drawing a sample $\theta_{i,T_i(t-1)}(\boldsymbol{a})$ from the posterior distribution. Formally, we have:

$$u_{i,T_i(t-1)}(\boldsymbol{a}) := \hat{\mu}_{i,T_i(t-1)}(\boldsymbol{a}) + \sqrt{b_{i,T_i(t-1)}}\hat{\sigma}_{i,T_i(t-1)}(\boldsymbol{a}), \tag{3}$$

$$\theta_{i,T_i(t-1)}(\boldsymbol{a}) \sim \mathcal{N}\left(\hat{\mu}_{i,T_i(t-1)}(\boldsymbol{a}), \hat{\sigma}^2_{i,T_i(t-1)}(\boldsymbol{a})\right), \tag{4}$$

where $b_{i,T_i(t-1)}$ is any non-negative sequence of values. From now on, we will refer to the version of the GCB algorithm that uses upper confidence bounds as GCB-UCB and the one resorting to sampling as GCB-TS (since it takes inspiration from the Thompson Sampling).

With either the values computed with Equation (3) or (4), the algorithm generates the estimated payoff vector $\bar{\boldsymbol{\mu}}_t := (\bar{\mu}_{1,T_i(t-1)}(\boldsymbol{a}_{11}), \ldots, \bar{\mu}_{C,T_i(t-1)}(\boldsymbol{a}_{CM_C}))$, $\bar{\boldsymbol{\mu}}_t \in \mathbb{R}^M$ (Line 7) either using the upper bounds $\bar{\mu}_{i,T_i(t-1)}(\boldsymbol{a}) := u_{i,T_i(t-1)}(\boldsymbol{a})$ or the samples from the distributions $\bar{\mu}_{i,T_i(t-1)}(\boldsymbol{a}) := \theta_{i,T_i(t-1)}(\boldsymbol{a})$. Then, the algorithm runs the (α, β)-*Approximation Oracle* on the estimated payoff vector $\bar{\boldsymbol{\mu}}_t$ to obtain the super-arm S_t to play in the next round (Line 8). Finally, the algorithm observes the payoffs $y_t(\mathbf{s}_{th})$ for each $\mathbf{s}_{th} \in S_t$ (Line 9), and updates the payoffs $\boldsymbol{y}(A_{i,T_i(t-1)})$ and selected arms $A_{i,T_i(t-1)}$ sequences (Line 11).

5. Finite-Time Regret Analysis

We show that the worst-case pseudo-regret of the GCB algorithm is upper bounded as follows. The proofs are reported in Appendix A.1 and A.2.

Theorem 1 *Given $\delta \in (0, 1)$, if we set $b_{i,n} := 2\log\left(\frac{CNM_i\pi_n}{\delta}\right)$, π_n being a sequence s.t. $\sum_{t=1}^{\infty}\frac{1}{\pi_t} = 1$ and $\pi_t > 0$, and given an (α, β)-Approximation Oracle, the pseudo-regret of the GCB-UCB algorithm $\mathcal{R}_N(\mathfrak{U})$ running on a GP-CMAB problem over N rounds is upper bounded with probability at least $1 - \delta$ by $\sqrt{\bar{c}\; CNB_{NM}\sum_{i=1}^{C}\gamma_{i,NM_i}}$, where $B_n := 2\log\left(\frac{CNM\pi_n}{\delta}\right)$, and $\bar{c} := \frac{8\Lambda^2}{\log(1+\sigma^{-2})}$.*

Theorem 2 *Given $\delta \in (0, 1)$ and an (α, β)-Approximation Oracle, the pseudo-regret of the GCB-TS $\mathcal{R}_N(\mathfrak{U})$ algorithm running on a GP-CMAB problem over N rounds is upper bounded with probability at least $1 - \delta$ by $\sqrt{\bar{c}\; CNB'_{NM}\sum_{i=1}^{C}\gamma_{i,NM_i}}$, where $B'_n := 8\log\left(\frac{2CNM\pi_n}{\delta}\right)$, π_n is a sequence such that $\sum_{t=1}^{\infty}\frac{1}{\pi_t} = 1$ and $\pi_t > 0$, and $\bar{c} := \frac{2(\alpha\beta+1)^2\Lambda^2}{\log(1+\sigma^{-2})}$.*

The upper bounds provided by Theorems 1 and 2 are expressed in terms of the maximum information gain γ_{i,NM_i} one might obtain over the different GPs. The problem of bounding such terms has been already discussed by (Srinivas et al., 2010), where the authors present the bounds for *linear kernel* $\gamma_{i,t} = \mathcal{O}(d\log t)$; *squared exponential kernel* $\gamma_{i,t} = \mathcal{O}((\log t)^{(d+1)})$; *Matérn kernel* $\gamma_{i,t} = \mathcal{O}(t^{d(d+1)/(2\nu+d(d+1))}\log t)$, if $\nu > 1$.

Our GCB algorithm suffers from a sublinear pseudo-regret when the above bounds on the information gain are employed. For instance, in the case of the squared exponential kernel, the term $\sum_{i=1}^{C}\gamma_{i,NM_i}$ in Theorems 1 and 2 can be bounded by $\mathcal{O}(C\log(NM)^{(d+1)})$. If we choose $\pi_n \propto \frac{1}{n^2}$, we obtain a pseudo-regret upper bound of:

$$\mathcal{R}_N(\mathfrak{U}) = \mathcal{O}\left(C\sqrt{N \log{(CN^3M^3)}(\log{(NM)})^{(d+1)}}\right).$$

Note that the bound scales linearly in the number of subsets C, but only logarithmically in the number of arms M. Therefore, our algorithm is suitable for those problems in which M is large, but they can be grouped into a few subsets (a single one in the best case) in which the arm payoffs are correlated.

6. Related Works

(Nuara et al., 2018) propose a CMAB algorithm, called AdComb, which employs GPs for the online optimization of Internet advertising campaigns. The AdComb algorithm is a specific case of our GCB algorithm. While an experimental evaluation of AdComb algorithm is provided, showing the different performance of the frequentist and Bayesian versions, no theoretical guarantees are known. (Degenne and Perchet, 2016) propose an algorithm for the setting in which a sub-Gaussian correlation among the payoff realizations exists. Their results are not comparable with the one provided here, in which the correlation is among expected payoffs.

Other related works on CMAB, not exploiting forms of correlation, are the following. (Chen et al., 2013) propose CUCB, which relies on statistical upper confidence bounds for bounded domain payoffs, showing an upper bound of $\mathcal{O}(M \log N)$. Our algorithm also applies to problems in which there is not any known upper bound on the payoffs and suffers from a worse pseudo-regret in terms of N and a better pseudo-regret in terms of M; this allows for its employment in challenging problems with a huge number of arms. (Ontañón, 2017) and (Shleyfman et al., 2014) propose the *naïve sampling* and LSI algorithms, respectively, to tackle a specific CMAB setting in which the reward function of a super-arm is a linear combination of the arms. While the former algorithm suffers from a pseudo-regret $\mathcal{O}(N)$, for the latter one no theoretical guarantee is provided.

Other works related to ours can be found in the (non-combinatorial) MAB literature. More precisely, (Srinivas et al., 2010) propose the GP-UCB algorithm that employs GPs in a stochastic MAB setting. The pseudo-regret of the algorithm is proved to be upper bounded in high probability as $\tilde{\mathcal{O}}(\sqrt{N})$. Our GCB algorithm extends this work to the more challenging combinatorial settings and presents the same upper bound on the pseudo-regret $\tilde{\mathcal{O}}(\sqrt{N})$. Instead, the Bayesian version of our GCB algorithm, when applied to (non-combinatorial) MAB settings, is the first Bayesian MAB algorithm employing GPs.

7. Conclusions

In this paper, we present GCB: an algorithm able to exploit the correlation among the expected payoffs of the arms in a GP-CMAB setting. This algorithm makes use of the existing GP structure to spread the information provided by the sampled payoff over the arm space. We provided a finite time analysis of the GCB algorithm, giving a high probability upper bound on the pseudo-regret of order $\tilde{\mathcal{O}}(C\sqrt{N})$, where C is the number of subsets presenting the GP structure and N is the time horizon.

Some interesting future works will concern the introduction of monotonicity assumptions for some of the input variables to tighten the bound, as well as the generalization of the results for continuous arm spaces. At last, an experimental campaign should be performed to evaluate the empiric performance of the two flavors of the GCB algorithm.

References

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1): 1–122, 2012.

K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 08 1995.

W. Chen, Y. Wang, and Y. Yuan. Combinatorial multi-armed bandit: General framework, results and applications. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 151–159, 2013.

Rémy Degenne and Vianney Perchet. Combinatorial semi-bandit with known covariance. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2972–2980, 2016.

Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. In *Proceedings of the IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages 1–9. IEEE, 2010.

C. Ko, J. Lee, and M. Queyranne. An exact algorithm for maximum entropy sampling. *Operations Research*, 43(4):684–691, 1995.

S. Maghsudi and E. Hossain. Multi-armed bandits with application to 5g small cells. *IEEE Wireless Communications*, 23(3):64–73, 2016.

Alessandro Nuara, Francesco Trovò, Nicola Gatti, and Marcello Restelli. A combinatorial-bandit algorithm for the online joint bid/budget optimization of pay-per-click advertising campaigns. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*, 2018.

Santiago Ontañón. Combinatorial multi-armed bandits for real-time strategy games. *Journal of Artificial Intelligence Research*, 58:665–702, 2017.

C. E. Rasmussen and C. K. Williams. *Gaussian Processes for Machine Learning*, volume 1. MIT press Cambridge, 2006.

Alexander Shleyfman, Antonín Komenda, and Carmel Domshlak. On combinatorial actions and cmabs with linear side information. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 825–830, 2014.

Niranjan Srinivas, Andreas Krause, Matthias Seeger, and Sham M Kakade. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1015–1022, 2010.

Appendix A. Appendix: Proofs

In what follows we provide the proofs of the two theorems we provided in the main paper.

A.1 Proof of Theorem 1

Proof This proof is partially inspired by the proof of Theorem 1 in (Srinivas et al., 2010). Be $r \sim \mathcal{N}(0,1)$ and $c \in \mathbb{R}^+$, we have:

$$\begin{aligned} \mathbb{P}[r > c] &= \frac{1}{\sqrt{2\pi}} e^{-\frac{c^2}{2}} \int_c^{\infty} e^{-\frac{(r-c)^2}{2} - c(r-c)} \, dr \\ &\leq e^{-\frac{c^2}{2}} \mathbb{P}[r > 0] = \frac{1}{2} e^{-\frac{c^2}{2}}, \end{aligned}$$

since $e^{-c(r-c)} \leq 1$ for $r \geq c$. For the symmetry of the Gaussian distribution, we have:

$$\mathbb{P}[|r| > c] \leq e^{-\frac{c^2}{2}}. \tag{5}$$

Given a generic sequence of elements $A_{i,t}$ coming from a single subset $\mathcal{D}_i$ and a corresponding sequence of payoffs $\boldsymbol{y}(A_{i,t})$, we have that $\mu_{ij} \sim \mathcal{N}(\hat{\mu}_{i,t}(\boldsymbol{a}_{ij}), \hat{\sigma}^2_{i,t}(\boldsymbol{a}_{ij}))$. Thus, once substituted $r = \frac{\mu_{ij} - \hat{\mu}_{i,t}(\boldsymbol{a}_{ij})}{\hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})}$ and $c = \sqrt{b_{i,t}}$ in Equation (5), we obtain:

$$\mathbb{P}\left[|\mu_{ij} - \hat{\mu}_{i,t}(\boldsymbol{a}_{ij})| > \sqrt{b_{i,t}} \hat{\sigma}_{i,t}(\boldsymbol{a}_{ij}) \right] \leq e^{-\frac{b_{i,t}}{2}}. \tag{6}$$

In a GP-CMAB setting, after n rounds, each arm can be chosen a number of times from 1 to n, therefore $1 \leq T_i(n) \leq nM_i$. Applying the union bound over the rounds ($n \in \{1, \dots, N\}$), the subsets of $\mathcal{D}$ ($\mathcal{D}_i$ with $i \in \{1, \dots, C\}$), the number of times the arms in $\mathcal{D}_i$ are chosen ($t \in \{1, \dots, nM_i\}$) and the available arms in $\mathcal{D}_i$ ($\boldsymbol{a}_{ij} \in \mathcal{D}_i$), and exploiting Equation (6), we obtain:

$$\mathbb{P}\left[\bigcup_{n,i,t,\boldsymbol{a}_{ij}} \left(|\mu_{ij} - \hat{\mu}_{i,t-1}(\boldsymbol{a}_{ij})| > \sqrt{b_{i,t}} \hat{\sigma}_{i,t-1}(\boldsymbol{a}_{ij}) \right) \right] \tag{7}$$

$$\leq \sum_{n=1}^{N} \sum_{i=1}^{C} \sum_{t=1}^{nM_i} M_i e^{-\frac{b_{i,t}}{2}}. \tag{8}$$

Thus, choosing $b_{i,t} = 2 \log \left(\frac{CNM_i \pi_t}{\delta} \right)$, we obtain:

$$\begin{aligned} \sum_{n=1}^{N} \sum_{i=1}^{C} \sum_{t=1}^{nM_i} M_i e^{-\frac{b_{i,t}}{2}} &\leq \sum_{i=1}^{C} \sum_{t=1}^{NM_i} \sum_{n=1}^{N} M_i e^{-\frac{b_{i,t}}{2}} \\ &= \sum_{i=1}^{C} \sum_{t=1}^{NM_i} NM_i \frac{\delta}{CNM_i \pi_t} \leq \delta \sum_{i=1}^{C} \left(\frac{1}{C} \sum_{t=1}^{\infty} \frac{1}{\pi_t} \right) = \delta. \end{aligned}$$

Therefore, for each $n \geq 1$, we know that with probability at least $1 - \delta$ the following holds for all $\boldsymbol{a}_{ij} \in \mathcal{D}_i$, $i \in \{1, \dots C\}$ and $T_i(n-1) \in \{1, \dots, nM_i\}$:

$$|\mu_{ij} - \hat{\mu}_{i,T_i(n-1)}(\boldsymbol{a}_{ij})| \leq \sqrt{b_{i,T_i(n-1)}} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}_{ij}). \tag{9}$$

The instantaneous pseudo-regret reg_n at round n satisfies the following inequality:

$$\begin{aligned} reg_n &= \alpha\beta r^*_{\boldsymbol{\mu}} - r_{\boldsymbol{\mu}}(S_n) \\ &\leq \alpha\beta r^*_{\boldsymbol{\mu}} - r_{\bar{\boldsymbol{\mu}}_n}(S_n) + r_{\bar{\boldsymbol{\mu}}_n}(S_n) - r_{\boldsymbol{\mu}}(S_n). \end{aligned} \tag{10}$$

Let us focus on the term $r_{\bar{\boldsymbol{\mu}}_n}(S_n)$. The following holds with probability at least β:

$$r_{\bar{\boldsymbol{\mu}}_n}(S_n) \geq \alpha\, r^*_{\bar{\boldsymbol{\mu}}_n} \geq \alpha\, r_{\bar{\boldsymbol{\mu}}_n}(S^*_{\boldsymbol{\mu}}) \geq \alpha\, r_{\boldsymbol{\mu}}(S^*_{\boldsymbol{\mu}}) = \alpha\, r^*_{\boldsymbol{\mu}}, \tag{11}$$

where $S^*_{\boldsymbol{\mu}} \in \arg\max_{S \in \mathcal{S}}(r_{\boldsymbol{\mu}}(S))$ is the super-arm providing the optimum expected reward when the expected payoffs are $\boldsymbol{\mu}$. In Equation (11) we exploit the fact that we have an (α, β)-approximation oracle and the definition of $r^*_{\bar{\boldsymbol{\mu}}_n}$ and the monotonicity property of the expected reward (Assumption 1), being $(\bar{\boldsymbol{\mu}}_n)_{ij} \geq \mu_{ij}, \forall i, j$. Since the approximation oracle guarantees an α approximation with probability β, on average the expected reward is:

$$r_{\bar{\boldsymbol{\mu}}_n}(S_n) \geq \alpha\, \beta\, r^*_{\boldsymbol{\mu}} + (1-\beta)\epsilon \geq \alpha\, \beta\, r^*_{\boldsymbol{\mu}} \qquad \forall \epsilon \geq 0. \tag{12}$$

Plugging the result of Equation (12) into Equation (10), the first two terms cancel out and we get:

$$\begin{aligned} reg_n &\leq r_{\bar{\boldsymbol{\mu}}_n}(S_n) - r_{\boldsymbol{\mu}}(S_n) \\ &\leq r_{\bar{\boldsymbol{\mu}}_n}(S_n) - r_{\boldsymbol{\mu}_n}(S_n) + r_{\boldsymbol{\mu}_n}(S_n) - r_{\boldsymbol{\mu}}(S_n), \end{aligned} \tag{13}$$

where $\boldsymbol{\mu}_n := \left(\hat{\mu}_{1,T_1(n-1)}(\boldsymbol{a}_{11}), \ldots, \hat{\mu}_{C,T_C(n-1)}(\boldsymbol{a}_{CM_C})\right)$ is the vector composed of the estimated average payoffs for each arm $\boldsymbol{a} \in \mathcal{D}$. We use the Lipschitz property of the expected reward function (see Assumption 2) to bound the terms $(r_{\bar{\boldsymbol{\mu}}_n}(S_n) - r_{\boldsymbol{\mu}_n}(S_n))$ and $(r_{\boldsymbol{\mu}_n}(S_n) - r_{\boldsymbol{\mu}}(S_n))$ appearing in Equation (13) as follows:

$$\begin{aligned} r_{\bar{\boldsymbol{\mu}}_n}(S_n) - r_{\boldsymbol{\mu}_n}(S_n) &= \Lambda ||\bar{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n||_\infty \\ &= \Lambda \max_{i \in \{1,\ldots,C\}} \left(\sqrt{b_{i,T_i(n-1)}} \max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}) \right) && (14) \\ &\leq \Lambda \sqrt{B_{nM}} \max_{i \in \{1,\ldots,C\}} \left(\max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}) \right) && (15) \\ &\leq \Lambda \sqrt{B_{nM}} \sum_{i=1}^{C} \left(\max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}) \right), \text{ and} && (16) \\ r_{\boldsymbol{\mu}_n}(S_n) - r_{\boldsymbol{\mu}}(S_n) &\leq \Lambda ||\boldsymbol{\mu}_n - \boldsymbol{\mu}||_\infty \\ &\leq \Lambda \sqrt{B_{nM}} \sum_{i=1}^{C} \left(\max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}) \right). && (17) \end{aligned}$$

Equation (14) holds by the definition of $\bar{\boldsymbol{\mu}}_n$. In Equation (15), we exploit that $b_{i,T_i(n-1)} = 2 \log \left(\frac{CN \max_i M_i \pi_{T_i(n-1)}}{\delta} \right) \leq 2 \log \left(\frac{CNM\pi_{nM}}{\delta} \right) = B_{nM}$. Equation (16) holds since the maximum over a set is not greater than the sum of the elements of the set, if they are all non-negative. Finally, Equation (17) directly follows from Equation (9). Plugging Equations (16) and (17) into Equation (13), we obtain:

$$reg_n \leq 2\Lambda \sqrt{B_{nM}} \sum_{i=1}^{C} \left(\max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}) \right). \tag{18}$$

We need now to upper bound $\hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a})$. Consider a realization μ_i of a GP over $\mathcal{D}_i$ and recall that, thanks to Lemma 5.3 in (Srinivas et al., 2010), under the Gaussian assumption we can express the information gain provided by $\boldsymbol{y}(A_{i,t})$ corresponding to the sequence of arms $A_{i,t}$ as:

$$\boldsymbol{I}(\boldsymbol{y}(A_{i,t}) \,|\, \mu_i) = \frac{1}{2} \sum_{h=1}^{t} \log \left(1 + \sigma^{-2}\, \hat{\sigma}^2_{i,h-1}(\boldsymbol{a}_{i,h})\right). \tag{19}$$

Since $b_{i,h}$ is non-decreasing in h, we can write:

$$\begin{aligned}\hat{\sigma}^2_{i,h-1}(\boldsymbol{a}_{i,h}) &= \sigma^2 \left(\sigma^{-2}\, \hat{\sigma}^2_{i,h-1}(\boldsymbol{a}_{i,h})\right) \\ &\leq \frac{\log\left(1 + \sigma^{-2}\, \hat{\sigma}^2_{i,h-1}(\boldsymbol{a}_{i,h})\right)}{\log\left(1 + \sigma^{-2}\right)},\end{aligned} \tag{20}$$

since $s^2 \leq \frac{\sigma^{-2} \log(1+s^2)}{\log(1+\sigma^{-2})}$ for all $s \in [0, \sigma^{-1}]$, and $\sigma^{-2}\, \hat{\sigma}^2_{i,h-1}(\boldsymbol{a}_{i,h}) \leq \sigma^{-2}\, k(\boldsymbol{a}_{i,h}, \boldsymbol{a}_{i,h}) \leq \sigma^{-2}$. Since Equation (20) holds for any $\boldsymbol{a} \in \mathcal{D}_i$ and for any $i \in \{1, \dots C\}$, then it also holds for the arm $\boldsymbol{a}_{\max}$ maximizing the variance $\hat{\sigma}^2_{i,h-1}(\boldsymbol{a}_{i,h})$ in μ_i defined over $\mathcal{D}_i$. Thus, setting $\bar{c} = \frac{8\,\Lambda^2}{\log(1+\sigma^{-2})}$ and exploiting the Cauchy-Schwarz inequality, we obtain:

$$\begin{aligned}\mathcal{R}^2_N(\mathfrak{U}) &\leq N \sum_{n=1}^{N} reg_n^2 \\ &\leq 4\Lambda^2 N \sum_{n=1}^{N} B_{nM} \left[\sum_{i=1}^{C} \left(\max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a})\right)\right]^2 \\ &\leq 4\Lambda^2 N \sum_{n=1}^{N} \left[B_{NM} C \sum_{i=1}^{C} \max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}^2_{i,T_i(n-1)}(\boldsymbol{a})\right] \\ &\leq \bar{c}\, C N B_{NM} \sum_{i=1}^{C} \frac{1}{2} \sum_{n=1}^{N} \max_{\boldsymbol{a} \in \mathcal{D}_i} \log\left(1 + \sigma^{-2} \hat{\sigma}^2_{i,T_i(n-1)}(\boldsymbol{a})\right) \\ &= \bar{c}\, C\, N\, B_{NM} \sum_{i=1}^{C} \max_{A_{i,T_i(N)} | \boldsymbol{a}_{i,h} \in \mathcal{D}_i} \boldsymbol{I}(\boldsymbol{y}(A_{i,T_i(N)}) \,|\, \mu_i) \\ &\leq \bar{c}\, C\, N\, B_{NM} \sum_{i=1}^{C} \gamma_{i,NM_i}.\end{aligned}$$

We conclude the proof by taking the square root on both the r.h.s. and the l.h.s. of the last inequality. ■

A.2 Proof of Theorem 2

Proof Consider a sequence of arms $A_{i,t}$ and their corresponding payoff realizations $\boldsymbol{y}(A_{i,t})$. Replacing $r = \frac{\hat{\mu}_{i,t}(\boldsymbol{a}_{ij}) - \theta_{i,t}(\boldsymbol{a}_{ij})}{\hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})}$ and $c = \sqrt{b'_{i,t}}$ in Equation (5), where $b'_{i,t} := 8 \log\left(\frac{2CNM_i\pi_n}{\delta}\right)$, we obtain:

$$\mathbb{P}\left[|\hat{\mu}_{i,t}(\boldsymbol{a}_{ij}) - \theta_{i,t}(\boldsymbol{a}_{ij})| > \sqrt{b'_{i,t}} \hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})\right] \leq e^{-\frac{b'_{i,t}}{2}}.$$

By relying on the triangle inequality, we derive:

$$\begin{aligned}
&\mathbb{P}\left\{|\mu_{ij} - \theta_{i,t}(\boldsymbol{a}_{ij})| > \sqrt{b'_{i,t}}\hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})\right\} \\
&\leq \mathbb{P}\left[|\mu_{ij} - \hat{\mu}_{i,t}(\boldsymbol{a}_{ij})| + \right. \\
&\left. |\hat{\mu}_{i,t}(\boldsymbol{a}_{ij}) - \theta_{i,t}(\boldsymbol{a}_{ij})| > \sqrt{b'_{i,t}}\hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})\right] \\
&\leq \mathbb{P}\left[|\mu_{ij} - \hat{\mu}_{i,t}(\boldsymbol{a}_{ij})| > \frac{1}{2}\sqrt{b'_{i,t}}\hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})\right] + \\
&+ \mathbb{P}\left[|\hat{\mu}_{i,t}(\boldsymbol{a}_{ij}) - \theta_{i,t}(\boldsymbol{a}_{ij})| > \frac{1}{2}\sqrt{b'_{i,t}}\hat{\sigma}_{i,t}(\boldsymbol{a}_{ij})\right] \\
&\leq 2e^{-\frac{b'_{i,t}}{8}} = \delta.
\end{aligned}$$

Similarly to what done in Equations (7)-(8), applying the union bound over the rounds, the subsets of $\mathcal{D}$, the number of times the arms are chosen in $\mathcal{D}_i$, and the available arms, we have that the following holds with probability at least $1 - \delta$:

$$|\mu_{ij} - \theta_{i,T_i(n-1)}(\boldsymbol{a}_{ij})| \leq \sqrt{b'_{i,T_i(n-1)}}\hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a}_{ij}), \tag{21}$$

for all $\boldsymbol{a}_{ij} \in \mathcal{D}_i$, $i \in \{1, \dots C\}$ and $T_i(n-1) \in \{1, \dots, nM_i\}$. The instantaneous pseudo-regret reg_n at round n is:

$$\begin{aligned}
reg_n &= \alpha\beta r^*_{\boldsymbol{\mu}} - r_{\boldsymbol{\mu}}(\hat{S}_n) \\
&= \alpha\beta r^*_{\boldsymbol{\mu}} - \alpha\beta r_{\boldsymbol{\theta_n}}(S^*_{\boldsymbol{\mu}}) + \alpha\beta r_{\boldsymbol{\theta_n}}(S^*_{\boldsymbol{\mu}}) - r_{\boldsymbol{\mu}}(\hat{S}_n),
\end{aligned} \tag{22}$$

where $\boldsymbol{\theta_n} := (\theta_{1,T_1(n-1)}(a_{11}), \dots, \theta_{C,T_C(n-1)}(a_{CM_C}))$ is the vector of the drawn payoffs for the turn n.

Since Equation (12) holds even for GCB-TS, we have that $r_{\boldsymbol{\theta_n}}(\hat{S}_n) \geq \alpha\beta r_{\boldsymbol{\theta_n}}(S^*_{\boldsymbol{\mu}})$ and the instantaneous pseudo-regret in Equation (22) becomes:

$$\begin{aligned}
reg_n &\leq \alpha\beta\left(r_{\boldsymbol{\mu}}(S^*_{\boldsymbol{\mu}}) - r_{\boldsymbol{\theta_n}}(S^*_{\boldsymbol{\mu}})\right) + r_{\boldsymbol{\theta_n}}(\hat{S}_n) - r_{\boldsymbol{\mu}}(\hat{S}_n) \\
&\leq \alpha\beta\Lambda||\boldsymbol{\mu} - \boldsymbol{\theta_n}||_\infty + \Lambda||\boldsymbol{\theta_n} - \mu||_\infty \qquad (23) \\
&\leq (\alpha\beta + 1)\Lambda \max_{i \in \{1,\dots,C\}} \left(\sqrt{b'_{i,T_i(n)}} \max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a})\right). \qquad (24)
\end{aligned}$$

Equation (23) holds by Assumption 2. Equation (24) holds with probability at least $1 - \delta$ for Equation (21). Exploiting the definition of B'_n, we obtain:

$$reg_n \leq (\alpha\beta + 1)\Lambda\sqrt{B'_{nM}} \sum_{i=1}^{C} \left(\max_{\boldsymbol{a} \in \mathcal{D}_i} \hat{\sigma}_{i,T_i(n-1)}(\boldsymbol{a})\right),$$

which is equal to Equation (18) apart from constants. The part of the proof of Theorem 1 that follows Equation (18) can be applied here since it only requires that B'_n is monotonically non-decreasing in n and this property holds by definition of B'_n. As a result, we obtain the same bound of Theorem 1 apart from constants. ■