

Thompson Sampling for Bandits with Clustered Arms

Emil Carlsson*, Devdatt Dubhashi, Fredrik D. Johansson

Department of Computer Science and Engineering, Chalmers University
 {caremil, dubhashi, fredrik.johansson}@chalmers.se

Abstract

We propose algorithms based on a multi-level Thompson sampling scheme, for the stochastic multi-armed bandit and its contextual variant with linear expected rewards, in the setting where arms are clustered. We show, both theoretically and empirically, how exploiting a given cluster structure can significantly improve the regret and computational cost compared to using standard Thompson sampling. In the case of the stochastic multi-armed bandit we give upper bounds on the expected cumulative regret showing how it depends on the quality of the clustering. Finally, we perform an empirical evaluation showing that our algorithms perform well compared to previously proposed algorithms for bandits with clustered arms.

1 Introduction

In a bandit problem, a learner must iteratively choose from a set of N actions, also known as arms, in a sequence of T steps as to minimize the expected cumulative regret over the horizon T [Lai and Robbins, 1985]. Inherent in this setup is an exploration-exploitation tradeoff where the learner has to balance between exploring actions she is uncertain about in order to gain more information and exploiting current knowledge to pick actions that appears to be optimal.

In this work, we consider versions of the standard multi-armed bandit problem (MAB) and the contextual bandit with linear rewards (CB) where there is a clustering of the arms known to the learner. In the standard versions of these problems the cumulative regret scales with number of arms, N , which becomes problematic when the number of arms grows large [Bubeck and Cesa-Bianchi, 2012]. Given a clustering structure one would like to exploit it to remove the explicit dependence on N and replace it with a dependence on the given clustering instead. A motivating example is recommender systems in e-commerce where there may be a vast amount of products organized into a much smaller set of categories. Users may have strong preferences for certain categories which yields similar expected rewards for recommending products from the same category.

*Contact Author

Our Contributions. We propose algorithms based on a multi-level Thompson sampling [Thompson, 1933] scheme for the stochastic multi-armed bandit with clustered arms (MABC) and its contextual variant with linear expected rewards and clustered arms (CBC). For the MABC, we provide regret bounds for our algorithms which completely removes the explicit dependence on N in favor for a dependence on properties of the given clustering. We perform an extensive empirical evaluation showing both how the quality of the clustering affects the regret and that our algorithms are very competitive with recent algorithms proposed for MABC and CBC. Noteworthy is that the empirical evaluation shows that our algorithms still performs well even in the case where our theoretical assumptions are violated.

2 Stochastic Multi-armed Bandit with Clustered Arms

We consider the MABC. As in the standard MAB problem we have a set of arms \mathcal{A} of cardinality N . At each time step $t > 0$ the learner must pick an arm $a_t \in \mathcal{A}$ after which an instant stochastic reward, $r_t(a_t)$, drawn from some distribution, $r_t \sim \mathcal{D}_{a_t}$, with an unknown mean $\mathbb{E}_{\mathcal{D}_{a_t}}[r_t] = \mu_{a_t}$. The goal of the learner is to maximize its expected cumulative reward over a sequence of T time steps or equivalently, to minimize its expected cumulative regret $\mathbb{E}[R_T]$ w.r.t the optimal arm $a^* = \arg \max_{a \in \mathcal{A}} \mu_a$ in hindsight, $R_T := \sum_{t=1}^T r_t(a^*) - r_t(a_t)$.

In the MABC, the learner has, in addition, access to a clustering of the N arms which may be used to guide exploration. We will consider two types of clustering:

Disjoint Clusters The N arms are partitioned into a set of clusters \mathcal{K} such that each arm $a \in \mathcal{A}$ is associated to exactly one cluster.

Hierarchical Clustering The N arms are organized into a tree \mathcal{T} of depth L such that each arm is associated with a unique leaf of the tree.

We will show in Section 2.2 and 2.4 that when rewards are drawn from Bernoulli distributions, $r_t \sim \mathcal{B}(\mu_a)$, with unknown parameters μ_a , the learner can exploit the known clustering to greatly improve the expected cumulative regret compared to the regret achievable with no knowledge of the cluster structure (under certain assumptions on the quality of the clustering).

Algorithm 1 TSC

Require: \mathcal{A}, \mathcal{K}
 Set $S_0 = F_0 = 1$ for all a and C .
for $t = 1, \dots, T$ **do**
 For each cluster C sample $\theta_C \sim \text{Beta}(S_t(C), F_t(C))$
 and pick $C_t = \arg \max_{C \in \mathcal{K}} \theta_C$
 For each $a \in C_t$ sample $\theta_a \sim \text{Beta}(S_t(a), F_t(a))$.
 Play arm $a_t = \arg \max_{a \in C_t} \theta_a$ and collect reward r_t .
 Update $S_{t+1}(a_t) = S_t(a_t) + r_t$, $F_{t+1}(a_t) = F_t(a_t) + (1 - r_t)$.
 Update $S_{t+1}(C_t) = S_t(C_t) + r_t$ and $F_{t+1}(C_t) = F_t(C_t) + (1 - r_t)$.
end for

2.1 Thompson Sampling for MABC

In the celebrated Thompson sampling (TS) algorithm for MAB with Bernoulli distributed rewards a learner starts at time $t = 0$ with a prior belief $\text{Beta}(1, 1)$ over possible expected rewards, $\theta_a \in [0, 1]$, for each $a \in \mathcal{A}$. At time t , having observed $S_t(a)$ number of successful ($r = 1$) plays and $F_t(a)$ the number of unsuccessful ($r = 0$) plays of arm a , the learner's posterior belief over possible expected rewards for arm a is $\text{Beta}(S_t(a), F_t(a))$, where $S_0(a) = F_0(a) = 1$. At each time step t , the learner samples an expected reward for each arm $\theta_a \sim \text{Beta}(S_t(a), F_t(a))$ and then acts greedily w.r.t. the sample means, i.e. the learner plays the arm $a_t = \arg \max_{a \in \mathcal{A}} \theta_a$. Given a reward r_t the learner updates the posterior of the played arm a_t as $S_{t+1}(a_t) = S_t(a_t) + r_t$ and $F_{t+1}(a_t) = F_t(a_t) + (1 - r_t)$. The posteriors of the arms not played are not updated.

Given a clustering of the arms into a set of clusters \mathcal{K} , we introduce a natural two-level bandit policy based on TS, Algorithm 1. In addition to the belief for each arm a , $\text{Beta}(S_t(a), F_t(a))$, the learner also keeps a belief over possible expected rewards $\text{Beta}(S_t(C), F_t(C))$ for each cluster $C \in \mathcal{K}$. At each t , the learner first use TS to pick a cluster - that is, it samples $\theta_C \sim \text{Beta}(S_t(C), F_t(C))$ for each cluster $C \in \mathcal{K}$ and then considers the cluster $C_t = \arg \max_{C \in \mathcal{K}} \theta_C$. The learner then samples $\theta_a \sim \text{Beta}(S_t(a), F_t(a))$ for each $a \in C_t$ and plays the arm $a_t = \arg \max_{a \in C_t} \theta_a$. Given a reward r_t the learner updates the beliefs for a_t and C_t as follows $S_{t+1}(a_t) = S_t(a_t) + r_t$, $F_{t+1}(a_t) = F_t(a_t) + (1 - r_t)$, $S_{t+1}(C_t) = S_t(C_t) + r_t$ and $F_{t+1}(C_t) = F_t(C_t) + (1 - r_t)$.

We extended this two-level scheme to hierarchical clustering of depth L , by recursively applying TS at each level of the tree, in Algorithm 2. The learner starts at the root of the hierarchical clustering, \mathcal{T} , and samples an expected reward for each of the sub-trees, \mathcal{T}_1^i spanned by its children, $i = 1, \dots$, from $\text{Beta}(S_t(\mathcal{T}_1^i), F_t(\mathcal{T}_1^i))$. The learner now traverses down to the root of the sub-tree satisfying $\mathcal{T}_{1,t}^i = \arg \max_{\mathcal{T}_1^i} \theta_{\mathcal{T}_1^i}$. This scheme is recursively applied until the learner reaches a leaf, i.e. an arm a_t , which is played. Given a reward r_t , each belief along the path from the root to a_t is updated using a standard TS update.

Algorithm 1 and 2 are not restricted to Bernoulli distributed rewards and can be used for any reward distribution with support $[0, 1]$ or for unbounded rewards by using Gaussian

Algorithm 2 HTS

Require: \mathcal{A}, \mathcal{T}
 Set $S_0(\mathcal{T}_l^i) = F_0(\mathcal{T}_l^i) = 1$ for each sub-tree \mathcal{T}_l^i .
for $t = 1, \dots, T$ **do**
 Set $\mathcal{T}_t = \mathcal{T}$.
 while \mathcal{T}_t is not a leaf **do**
 For each sub-tree \mathcal{T}_l^i spanned by the children of \mathcal{T}_t
 sample $\theta_{\mathcal{T}_l^i} \sim \text{Beta}(S_t(\mathcal{T}_l^i), F_t(\mathcal{T}_l^i))$.
 Set $\mathcal{T}_t = \arg \max \theta_{\mathcal{T}_l^i}$.
 end while
 Play the arm a_t corresponding to the leaf \mathcal{T}_t and collect the reward r_t .
 Perform a TS update on each $S_t(\mathcal{T}_l^i), F_t(\mathcal{T}_l^i)$ on the path to a_t .
end for

prior and likelihood in TS, as done for the standard MAB in [Agrawal and Goyal, 2017].

2.2 Regret Analysis TSC

Assume that we have a clustering of N Bernoulli arms, into a set of clusters \mathcal{K} . For each arm a , let μ_a denote the expected reward and let a^* be the unique optimal arm with expected reward μ^* . We denote the cluster containing a^* as C^* . We denote the expected regret for each a as $\Delta_a := \mu^* - \mu_a$ and for each cluster $C \in \mathcal{K}$, we define $\bar{\mu}_C = \max_{a \in C} \mu_a$, $\underline{\mu}_C = \min_{a \in C} \mu_a$ and $\Delta_C = \mu^* - \bar{\mu}_C$.

For each cluster $C \in \mathcal{K}$ we define distance d_C to the optimal cluster C^* as $d_C = \min_{a \in C^*, \hat{a} \in C} \mu_a - \mu_{\hat{a}}$ and the width w_C as $w_C = \bar{\mu}_C - \underline{\mu}_C$, let w^* denote the width of the optimal cluster.

Assumption 1 (Strong Dominance). For $C \neq C^*$, $d_C > 0$.

This assumption is equivalent to what is referred to as *tight clustering* in [Bouneffouf *et al.*, 2019] and *strong dominance* in [Jedor *et al.*, 2019]. In words, we assume that, in expectation, every arm in the optimal cluster is better than every arm in any suboptimal cluster.

In order to bound the regret of TSC we will repeatedly use the following seminal result for the standard MAB case (without clustering) from [Kaufmann *et al.*, 2012]. Here, we denote the Kullback-Leibler divergence between two Bernoulli distributions with means μ_1 and μ_2 as $D(\mu_1, \mu_2)$ and the natural logarithm of T as $\log T$.

Theorem 1 ([Kaufmann *et al.*, 2012]). *In the standard multi-arm bandit case with optimal arm reward μ^* , the number of plays of a sub-optimal arm a using TS is bounded from above, for any $\epsilon > 0$, by*

$$(1 + \epsilon) \frac{1}{D(\mu_a, \mu^*)} (\log T + \log \log T) + O(1).$$

Our plan is to apply Theorem 1 in two different cases: to bound the number of times a sub-optimal cluster is played and to bound the number of plays of a sub-optimal arm in the optimal cluster. However, the theorem not directly applicable to the number of plays of a sub-optimal cluster, $N_{C,T}$, since the reward distribution is drifting as the policy is learning about

the arms within C . Nevertheless, we can use a comparison argument to bound the number of plays of a sub-optimal cluster by plays in an auxiliary problem with stationary reward distributions and get the following lemma.

Lemma 2. *For any $\epsilon > 0$ and assuming strong dominance, the expected number of plays of a sub-optimal cluster C at time T using TSC is bounded from above by*

$$E[N_{C,T}] \leq \frac{1 + \epsilon}{D(\bar{\mu}_C, \underline{\mu}_{C^*})} (\log T + \log \log T) + O(1).$$

We can use Lemma 2 to derive the following instance-dependent regret bound for TSC.

Theorem 3. *For any $\epsilon > 0$, the expected regret of TSC under the assumption of strong dominance is bounded from above by*

$$(1 + \epsilon) \left(\sum_{C \neq C^*} \frac{\Delta_C}{D(\bar{\mu}_C, \underline{\mu}_{C^*})} + \sum_{a \in C^*} \frac{\Delta_a}{D(\mu_a, \mu^*)} \right) \log T + o(\log T).$$

We can derive an instance-independent upper bound from Theorem 3 which only depends on number of clusters, number of arms in the optimal cluster and the quality of the clustering. Now, define γ_C as the ratio between width of the optimal cluster and the distance of C to the optimal cluster:

$$\gamma_C := \begin{cases} w^*/d_C, & C \neq C^* \\ 0, & \text{otherwise} \end{cases}$$

and let $\gamma := \sum_C \gamma_C / K$. We arrive at the following result.

Theorem 4. *Assume strong dominance and let A^* be the number of arms in the optimal cluster and K the number of sub-optimal clusters. The expected regret of TSC is bounded from above by $\mathbb{E}[R_T] \leq O\left(\sqrt{(A^* + K(1 + \gamma))T \log T}\right)$.*

Clustering Quality and Regret. As a sanity check, we note that if the expected rewards of all arms in the optimal cluster are equal we have $\gamma = 0$ and the bound in Theorem 4 reduces to the bound for the standard MAB in [Agrawal and Goyal, 2017] with $K + 1$ arms. On the other hand, if the optimal cluster has a large width along with many sub-optimal clusters with a small distance to the optimal cluster γ becomes large and little is gained from the clustering. Two standard measures of cluster quality are the (a) the maximum diameter/width of a cluster and (b) inter-cluster separation. We see that for our upper bound, *only the width of the optimal cluster and the separation of other clusters from the optimal cluster* are important. These dependencies are consistent with the observations in [Pandey *et al.*, 2007, Section 5.3], which suggest that high cohesiveness within the optimal cluster and large separation are crucial for achieving low regret. However our analysis is more precise than their observations and we also provide rigorous regret bounds.

2.3 Lower Bounds for Disjoint Clustering

In the case of Bernoulli distributed rewards we can derive the following lower bound for the instance dependent case using the pioneering works of [Lai and Robbins, 1985].

Theorem 5. *The expected regret for any policy, on the class of bandit problems with Bernoulli distributed arms clustered such that strong dominance holds, is bounded from below by*

$$\liminf_{T \rightarrow \infty} \frac{\mathbb{E}[R_T]}{\log T} \geq \sum_{a \in C^*} \frac{\Delta_a}{D(\mu_a, \mu^*)} + \sum_{C \neq C^*} \frac{\Delta_C}{D(\underline{\mu}_C, \mu^*)}$$

We compare the lower bound in Theorem 5 to our instance-dependent upper bound in Theorem 3 and we see that the regret suffered in TSC from playing sub-optimal clusters asymptotically differs from the corresponding term in the lower bound by a factor depending on the width of the clusters since

$$D(\bar{\mu}_C, \underline{\mu}_{C^*}) = D(\underline{\mu}_C + w_C, \mu^* - w^*) \leq D(\underline{\mu}_C, \mu^*).$$

Thus, as the width of the clusters goes to zero, the regret of TSC approaches the lower bound. However, as also discussed in [Jedor *et al.*, 2019] it is unclear whether the lower bound derived in Theorem 5 can be matched by any algorithm since it doesn't depend at all on the quality on the given clustering and assumes the optimal policy to always play the worst action in sub-optimal clusters.

The following minimax lower bound follows trivially from the $\Omega(\sqrt{NT})$ minimax lower bound for standard MAB [Auer *et al.*, 1998] by considering the two cases: where all clusters are singletons and all arms are in one cluster.

Theorem 6. *Let K be the number of sub-optimal clusters and let A^* be the number of arms in the optimal cluster. The expected regret for any policy, on the class of bandit problems with Bernoulli distributed arms clustered such that strong dominance holds, satisfies $\mathbb{E}[R_T] \geq \Omega(\sqrt{(A^* + K)T})$.*

Let $d > 0$ be the smallest distance between any sub-optimal and the optimal cluster. We compare Theorem 6 to the upper bound in Theorem 4 and observe that $\sqrt{(A^* + K)T} \leq \sqrt{(A^* + (1 + \gamma)K)T} \leq \sqrt{(1 + \frac{1}{d})} \sqrt{(A^* + K)T}$. Hence, our upper bound in Theorem 4 matches the lower bound up to logarithmic factors and a constant depending on the separation of the clusters.

2.4 Regret Analysis HTS

Assume we have N Bernoulli arms clustered into a tree \mathcal{T} and for simplicity we assume it to be perfectly height-balanced. We denote the sub-tree corresponding to node j on level i as \mathcal{T}_i^j and on each level i we denote the sub-tree containing the optimal arm as \mathcal{T}_i^* . Let \mathcal{T}_{i+1}^j , $j \in [1, K_i^*]$, denote sub-trees spanned by the child nodes of the root in \mathcal{T}_i^* , where K_i^* is the number of children of the root in \mathcal{T}_i^* . W.l.o.g let $j = 1$ be the sub-tree, \mathcal{T}_{i+1}^1 , that contains the optimal action. For each sub-tree \mathcal{T}_i^j we define $\Delta_i^j := \mu^* - \max_{a \in \mathcal{T}_i^j} \mu_a$ and $d_i^j := \min_{a \in \mathcal{T}_i^*} \mu_a - \max_{a \in \mathcal{T}_i^j} \mu_a$.

Assumption 2 (Hierarchical Strong Dominance). *We assume $d_i^j > 0$, $\forall i, j$ except for \mathcal{T}_i^* .*

Under this assumption the results in Theorem 3 can be naturally extended to HTS by recursively applying Theorem 3.

Theorem 7. Assuming hierarchical strong dominance. For any $\epsilon > 0$, the expected regret of HTS is upper bounded by

$$(1 + \epsilon) \left(\sum_{i=0}^{L-1} \sum_{j=2}^{K_i} \frac{\Delta_j^i}{(d_j^i)^2} + \sum_{a \in \mathcal{T}_L^*} \frac{1}{\Delta_a} \right) \log T + o(\log T).$$

For $L = 0$ Theorem 2.4 reduces to the instance-dependent bound for standard TS and for $L = 1$ it reduces to the bound for TSC presented in Theorem 3. Hierarchical structures and bandits have previously been studied in the prominent works [Coquelin and Munos, 2007] and [Bubeck *et al.*, 2011] which assumes there is a known smoothness. Here we do not make such assumptions and Theorem 7 instead relies on an assumption regarding the ordering of the tree.

Plausibility of Hierarchical Strong Dominance. The hierarchical strong dominance assumption is perhaps too strong for a general hierarchical clustering but it might be reasonable for shallow trees. One example is in e-commerce where products can be organized into sub-categories and later categories. A user might have a strong preference for the sub-category “Football” in the category “Sports”.

3 Contextual Bandit with Linear Rewards and Clustered Arms

In this section, we consider the MABC problem in its contextual variant with linear expected rewards (CBC). As in the classic CB, there is for each arm $a \in \mathcal{A}$ an, a priori, unknown vector $\theta_a \in \mathbf{R}^d$. At each time t , the learner observes a context vector $x_t \in \mathbf{R}^d$ and the expected reward for each arm a at time t , given that the learner has observed x_t , is $\mathbb{E}[r_t(a)|x_t] = x_t^\top \theta_a$. Similar to MABC, the learner has, in addition, access to a clustering of the N arms and for CBC we assume the arms to be clustered into a set of \mathcal{K} disjoint clusters.

For the CBC we extend TSC, Algorithm 1, to LinTSC, as defined in Algorithm 3. At each level of LinTSC, we use the Thompson sampling scheme developed for standard CB in [Agrawal and Goyal, 2012].

Algorithm 3 LinTSC

Require: $v > 0$

Set $B_c = \mathbf{1}_d, f_c = \mathbf{0}_d, \mu_c = \mathbf{0}_d, B_{c,i} = \mathbf{1}_d, f_{c,i} = \mathbf{0}_d, \mu_{c,i} = \mathbf{0}_d$

for $t = 1, \dots, T$ **do**

Observe context x_t

Sample $\theta_c \sim \mathcal{N}(\mu_c^\top x_t, v x_t^\top B_c^{-1} x_t)$

Consider cluster $k = \arg \max_c \theta_c$

Sample $\theta_{k,i} \sim \mathcal{N}(\mu_{k,i}^\top x_t, v x_t^\top B_{k,i}^{-1} x_t)$

Play arm $a = \arg \max_i \theta_{k,i}$

Observe reward r_t and update $B_k = B_k + x_t x_t^\top, B_{k,a} = B_{k,a} + x_t x_t^\top, f_k = f_k + r x_t, f_{k,i} = f_{k,i} + r x_t, \mu_k = B_k^{-1} f_k$ and $\mu_{k,i} = B_{k,i}^{-1} f_{k,i}$.

end for

4 Experimental Results

4.1 Stochastic Multi-armed Bandit

Strong Dominance. We generate synthetic data, for which strong dominance holds, in the following way: We have N arms and each arm i is Bernoulli distributed with reward probability p_i . The arms are clustered into K clusters and we have A^* arms in the optimal cluster. For the remaining $N - A^*$ arms we assign each arm to one of the sub-optimal clusters with uniform probability. We set the reward probability of the best arm in the optimal cluster to be 0.6 and for the worst arm in the optimal cluster we set it to be $0.6 - w^*$. For the remaining $A^* - 2$ arms in the optimal cluster we draw the reward probability from $\mathcal{U}(0.6 - w^*, 0.6)$ for each arm. In each sub-optimal cluster we set the probability of the best arm to be $0.6 - w^* - d$ and for the worst arm to be $0.5 - w^* - d$, the probability for the remaining arms are drawn from $\mathcal{U}(0.5 - w^* - d, 0.6 - w^* - d)$. The optimal cluster will then have a width of w^* and the distance from each sub-optimal cluster to the optimal cluster will be d . In Figures 1a–1e, we run TS and TSC on the same instances for $T = 3000$ time steps, varying the different instance parameters and plotting the cumulative regret of each algorithm at the final time step T . For each set of parameters we evaluate the algorithms using 50 different random seeds and the error bars corresponds to ± 1 standard deviation.

In Figures 1a and 1b, we observe that the cumulative regret scales depending on the clustering quality parameters d and w^* as suggested by our bounds in Section 2.2—that is, the cumulative regret of TSC decreases as d increases and increases as w^* increases. In Figure 1c, we observe that the linear dependence in N for TS is changed to a linear dependence in K and A^* , Figures 1d and 1e, which greatly reduces the regret of TSC compared to TS as the size of the problem instance increases. In Figure 1e we also see that as the number of arms in the optimal cluster, A^* , increases to be a substantial amount of the total number of arms, the gain from using TSC compared to TS vanishes.

Hierarchical Strong Dominance. We generate a bandit problem by first uniformly sample N Bernoulli arms from $\mathcal{U}(0.1, 0.8)$ followed by recursively sorting and merging the arms into a balanced binary tree, which has the hierarchical strong dominance property. In Figure 1f, we ran the algorithms for $T = 3000$ over 50 random seeds and illustrated how the cumulative regret at time T of HTS changes as we alter the depth L of the given tree and the total number of arms N . Note that $L = 0$ corresponds to TS and $L = 1$ corresponds to TSC. We observe that as the size of the problem instance grows, i.e increasing N , using more levels in the tree becomes more beneficial due to aggressive exploration scheme of HTS. Hence, once we realize that one sub-tree is better than the other we discard all arms in the corresponding sub-optimal sub-tree. Connecting back to Theorem 7 we see that HTS gets only a dependence $O(\log_2 N)$ in the number of arms when using the full hierarchical tree in Figure 1f.

Violation of Assumptions. In a real world setting, assuming that strong dominance and especially hierarchical strong dominance holds completely is often too strong. We thus

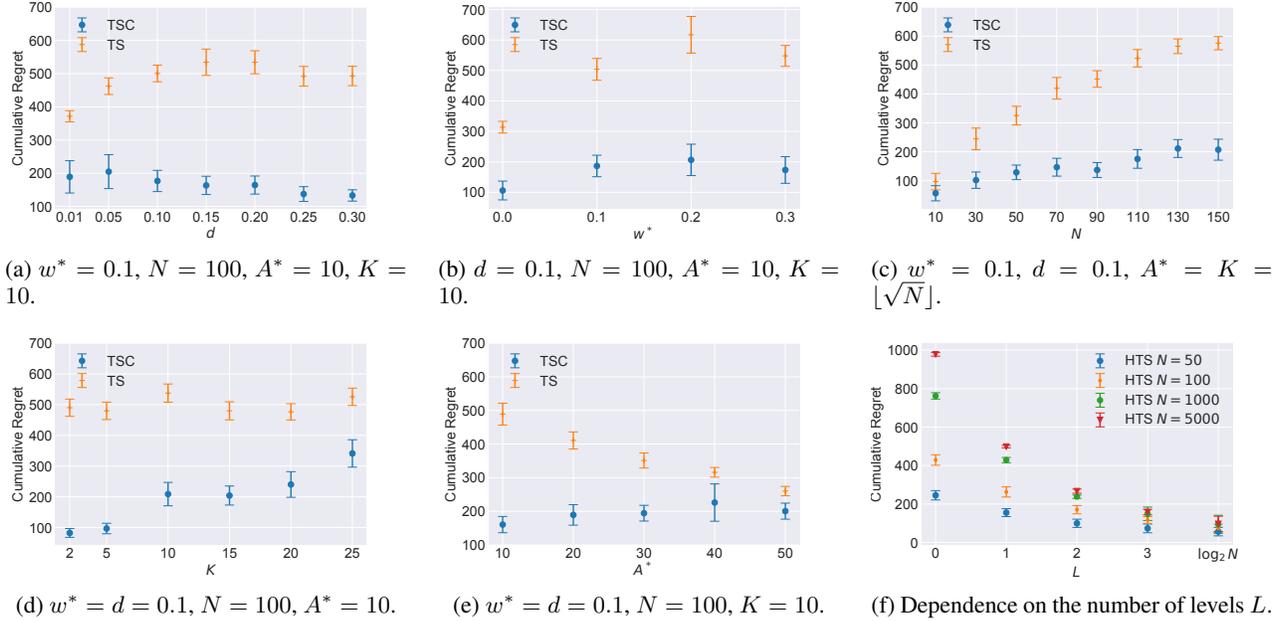


Figure 1: Strong and Hierarchical Strong Dominance.

evaluate our algorithms on instances for which these assumptions are violated. We generate N arms by for each arm i we sample a value $x_i \sim \mathcal{U}(0, 1)$. We cluster the arms into K clusters, based on the values $\{x_i\}$, using K-means. The reward distribution of each arm i is a Bernoulli distribution with mean $f(x_i)$ where $f(x) = \frac{1}{2}(\sin 13x \sin 27x + 1)$. This function is illustrated in the supplementary material, Appendix A, and has previously been used to evaluate bandit algorithms in [Bubeck *et al.*, 2011], the smoothness of the function ensures arms within the same cluster to have similar expected rewards, on the other hand the periodicity of sin yields many local optima and the optimal cluster won't strongly dominate the other clusters. On these instances, we benchmark TSC against two another algorithms proposed for MABC, UCBC [Pandey *et al.*, 2007, Bouneffouf *et al.*, 2019] and TSMAX [Zhao *et al.*, 2019]. We also benchmark against UCB1 [Auer *et al.*, 2002] and TS which both considers the problem as a standard MAB, making no use of the clustering. We run the algorithms on two different instances, one with $N = 100$ and $K = 10$ and the other one with $N = 1000$ and $K = 32$. For each instance we run the algorithms on 100 different random seeds and we present the results in Figure: 2a and 2b, the error bars corresponds to ± 1 standard deviation. TSC outperforms the other algorithms on both instances and especially on the larger instance where there is a big gap between the regret of TSC and the regret of the other algorithms. In order to test HTS we generate an instance, as above, with $N = 5000$ and $K = 15$ and construct a tree by recursively breaking each cluster up into 15 smaller clusters using k-means. In Figure 2c we show the performance of HTS for two different levels, $L = 2, 3$, compared to TSC using the clusters at level $L = 1$ in the tree and also compared to the UCT-algorithm [Kocsis and Szepesvári, 2006] using the same levels of the

tree as HTS. We averaged over 100 random seeds. The HTS performs well on this problem and is slightly better than TSC while both HTS and TSC outperforms UCT. We present more empirical results for MABC in the supplementary material.

4.2 Contextual Bandit

We generate contextual data in the same way as in [Bouneffouf *et al.*, 2019]. We have K clusters and N arms. Each arm j is randomly assigned to a cluster i . For each cluster i we sample a centroid $\theta_i^c \sim \mathcal{N}(0, \mathbf{I}_5)$ and define a coefficient for each arm j in the cluster as $\theta_j = \theta_i^c + \epsilon v$, $v \sim \mathcal{N}(0, \mathbf{I}_5)$. We take the reward of an arm to be $\mathcal{U}(0, 2\theta_j^\top x)$ where x is the given context. The reward becomes linear and we can control the expected diameter of a cluster by varying ϵ .

We benchmark LinTSC against the UCB-based counterpart LinUCBC [Bouneffouf *et al.*, 2019] and the standard algorithms LinTS [Agrawal and Goyal, 2012] and LinUCB [Li *et al.*, 2010], which treats the problem as a standard CB. We ran the algorithms on three different instances presented in Figures 2d, 2e and 2f, over 25 different random seeds and the error bars corresponds to ± 1 standard deviation. We run all algorithms with there corresponding standard parameter ($v = 1$ for LinTS and LinTSC, $c = 2$ for LinUCB and LinUCBC). We see a clear improvement between not using the clustering (TS) and using the clustering (TSC). LinTSC performs slightly better than LinUCBC as the problem becomes larger w.r.t number of arms and clusters, Figures 2e and 2f.

5 Related Work

Bandits are now a classical subject in machine learning and recent textbook treatments are [Bubeck and Cesa-Bianchi, 2012, Slivkins, 2019, Lattimore and Szepesvári, 2020]. The MABC and CBC can be considered as natural special cases

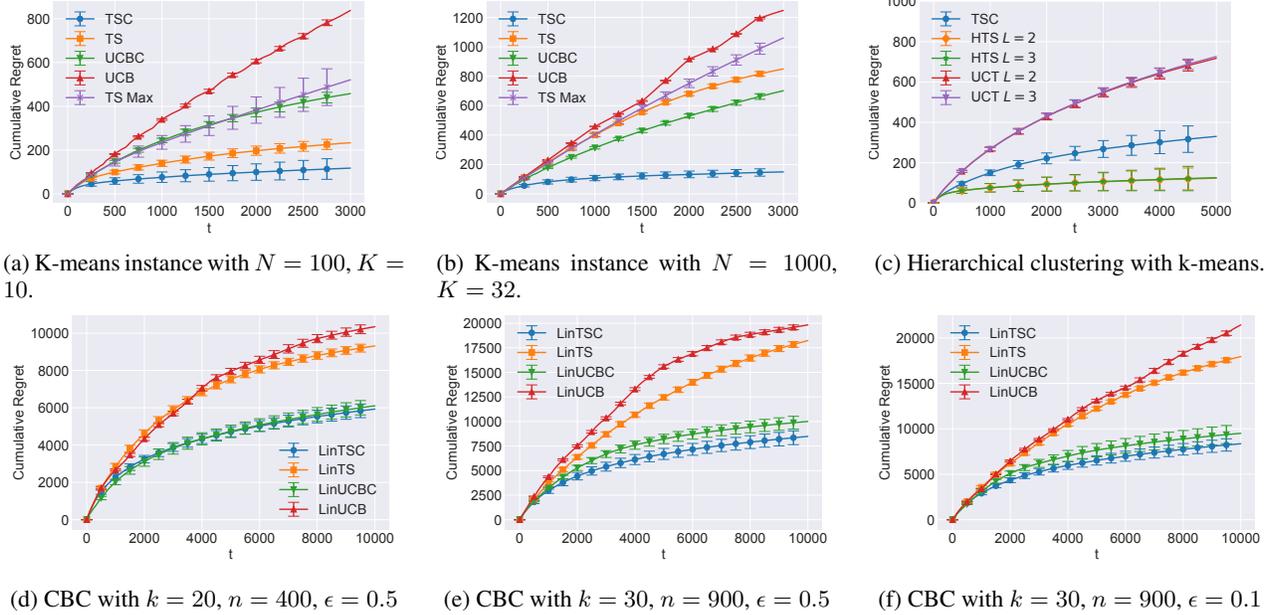


Figure 2: CBC and violation of assumptions in MABC.

of the more general finite-armed structured bandit which is studied in [Lattimore and Munos, 2014, Combes *et al.*, 2017, Gupta *et al.*, 2018, Gupta *et al.*, 2019]. To the best of our knowledge, the idea of clustered arms was first studied in [Pandey *et al.*, 2007] and the MABC corresponds to their undiscounted MDP setup for which the authors propose a general two-level bandit policy and gives theoretical justifications on how the regret scales depending on the characteristics of the clustering, but without stating rigorous regret bounds. Bandits with clustered arms were also recently studied in [Bouneffouf *et al.*, 2019, Jedor *et al.*, 2019] and both papers prove regret bounds for UCB-styled algorithms in the MABC under various assumptions on the clustering. [Bouneffouf *et al.*, 2019] is the work most related to ours since they consider a two-level UCB scheme and regret bounds that exhibits similar dependence on the clustering quality as our bounds. In [Zhao *et al.*, 2019] the authors propose a two-level TS algorithm where the belief of a cluster is set to the belief of the best performing arm in the cluster so far and the authors give no theoretical analysis of its regret. Clustered arms also appear in the regional bandit model [Wang *et al.*, 2018, Singh *et al.*, 2020] under the assumption that all arms in one cluster share the same underlying parameter. Another model related to our work is the latent bandit [Maillard and Mannor, 2014, Hong *et al.*, 2020] where the reward distributions depends on a latent state and the goal of the learner is to identify this state.

Bandits and tree structures are studied using a UCB-styled algorithm for Monte-Carlo-based planning in the influential work [Kocsis and Szepesvári, 2006] and later studied for various bandit problems with smoothness in the seminal works [Coquelin and Munos, 2007, Bubeck *et al.*, 2011].

We have based our bandit algorithms on the classical

method Thompson sampling [Thompson, 1933] which has been shown to perform well in practise [Chapelle and Li, 2011] and for which rigorous regret analyses recently have been established for the standard MAB in [Kaufmann *et al.*, 2012, Agrawal and Goyal, 2017]. The contextual version of Thompson sampling we use in our two-level scheme for CBC was originally proposed and analyzed for standard CB in [Agrawal and Goyal, 2012] and recently revisited in [Abeille and Lazaric, 2017].

6 Conclusions

In this paper, we have addressed the stochastic multi-armed bandit problem and the contextual bandit with clustered arms and proposed algorithms based on multi-level Thompson sampling. We have shown that our algorithms can be used to drastically reduce the regret when a clustering of the arms is known and that these algorithms are competitive to its UCB-based counterparts. We think that the simplicity of our algorithms and the fact that one can easily incorporate prior knowledge makes them well-suited options for bandit problems with a known clustering structure of the arms. In the future we would like to explore how the regret of TSC behaves under weaker assumptions on the clustering. We want to determine what are sufficient properties of the clustering to ensure sub-linear regret of LinTSC.

Acknowledgments

This work was supported by funding from CHAIR (Chalmers AI Research Center) and from the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations in this work were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC).

References

- [Abeille and Lazaric, 2017] Marc Abeille and Alessandro Lazaric. Linear Thompson Sampling Revisited. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 176–184, 20–22 Apr 2017.
- [Agrawal and Goyal, 2012] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. *30th International Conference on Machine Learning, ICML 2013*, 09 2012.
- [Agrawal and Goyal, 2017] Shipra Agrawal and Navin Goyal. Near-optimal regret bounds for thompson sampling. *J. ACM*, 64(5):30:1–30:24, 2017.
- [Auer et al., 1998] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. *Foundations of Computer Science, 1975., 16th Annual Symposium on*, 07 1998.
- [Auer et al., 2002] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2):235–256, 2002.
- [Bouneffouf et al., 2019] Djallel Bouneffouf, Srinivasan Parthasarathy, Horst Samulowitz, and Martin Wistuba. Optimal exploitation of clustering and history information in multi-armed bandit. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 2016–2022, 7 2019.
- [Bubeck and Cesa-Bianchi, 2012] Sébastien Bubeck and Nicolò Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.
- [Bubeck et al., 2011] Sébastien Bubeck, Remi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12, 05 2011.
- [Chapelle and Li, 2011] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. In *Advances in Neural Information Processing Systems 24*, pages 2249–2257. 2011.
- [Combes et al., 2017] Richard Combes, Stefan Magureanu, and Alexandre Proutiere. Minimal exploration in structured stochastic bandits. In *Advances in Neural Information Processing Systems 30*, pages 1763–1771. 2017.
- [Coquelin and Munos, 2007] Pierre-Arnaud Coquelin and Rémi Munos. Bandit Algorithms for Tree Search. In *Uncertainty in Artificial Intelligence*, 2007.
- [Gupta et al., 2018] S. Gupta, Gauri Joshi, and O. Yağan. Exploiting correlation in finite-armed structured bandits. *ArXiv*, abs/1810.08164, 2018.
- [Gupta et al., 2019] S. Gupta, S. Chaudhari, Gauri Joshi, and O. Yağan. Multi-armed bandits with correlated arms. *ArXiv*, abs/1911.03959, 2019.
- [Hong et al., 2020] Joey Hong, Branislav Kveton, Manzil Zaheer, Yinlam Chow, Amr Ahmed, and Craig Boutilier. Latent bandits revisited. In *Advances in Neural Information Processing Systems 33*, 2020.
- [Jedor et al., 2019] Matthieu Jedor, Vianney Perchet, and Jonathan Louedec. Categorized bandits. In *Advances in Neural Information Processing Systems*, volume 32, pages 14422–14432, 2019.
- [Kaufmann et al., 2012] Emilie Kaufmann, Nathaniel Korda, and Rémi Munos. Thompson sampling: An asymptotically optimal finite-time analysis. In *Algorithmic Learning Theory - 23rd International Conference, ALT 2012. Proceedings*, volume 7568, pages 199–213, 2012.
- [Kocsis and Szepesvári, 2006] Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *In: ECML-06. Number 4212 in LNCS*, pages 282–293, 2006.
- [Lai and Robbins, 1985] T.L. Lai and H Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.
- [Lattimore and Munos, 2014] Tor Lattimore and Remi Munos. Bounded regret for finite-armed structured bandits. In *Advances in Neural Information Processing Systems*, volume 27, pages 550–558, 2014.
- [Lattimore and Szepesvári, 2020] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.
- [Li et al., 2010] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. *Proceedings of the 19th international conference on World wide web - WWW '10*, 2010.
- [Maillard and Mannor, 2014] Odalric-Ambrym Maillard and Shie Mannor. Latent bandits. *31st International Conference on Machine Learning, ICML 2014*, 1, 05 2014.
- [Pandey et al., 2007] Sandeep Pandey, Deepayan Chakrabarti, and Deepak Agarwal. Multi-armed bandit problems with dependent arms. In *ICML*, pages 721–728, 2007.
- [Singh et al., 2020] Rahul Singh, Fang Liu, Yin Sun, and Ness Shroff. Multi-armed bandits with dependent arms. *arXiv*, 2020.
- [Slivkins, 2019] Aleksandrs Slivkins. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019.
- [Thompson, 1933] William 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.
- [Wang et al., 2018] Zhiyang Wang, Ruida Zhou, and Cong Shen. Regional multi-armed bandits. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84, pages 510–518, 2018.
- [Zhao et al., 2019] T. Zhao, M. Li, and M. Poloczek. Fast reconfigurable antenna state selection with hierarchical thompson sampling. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6, 2019.